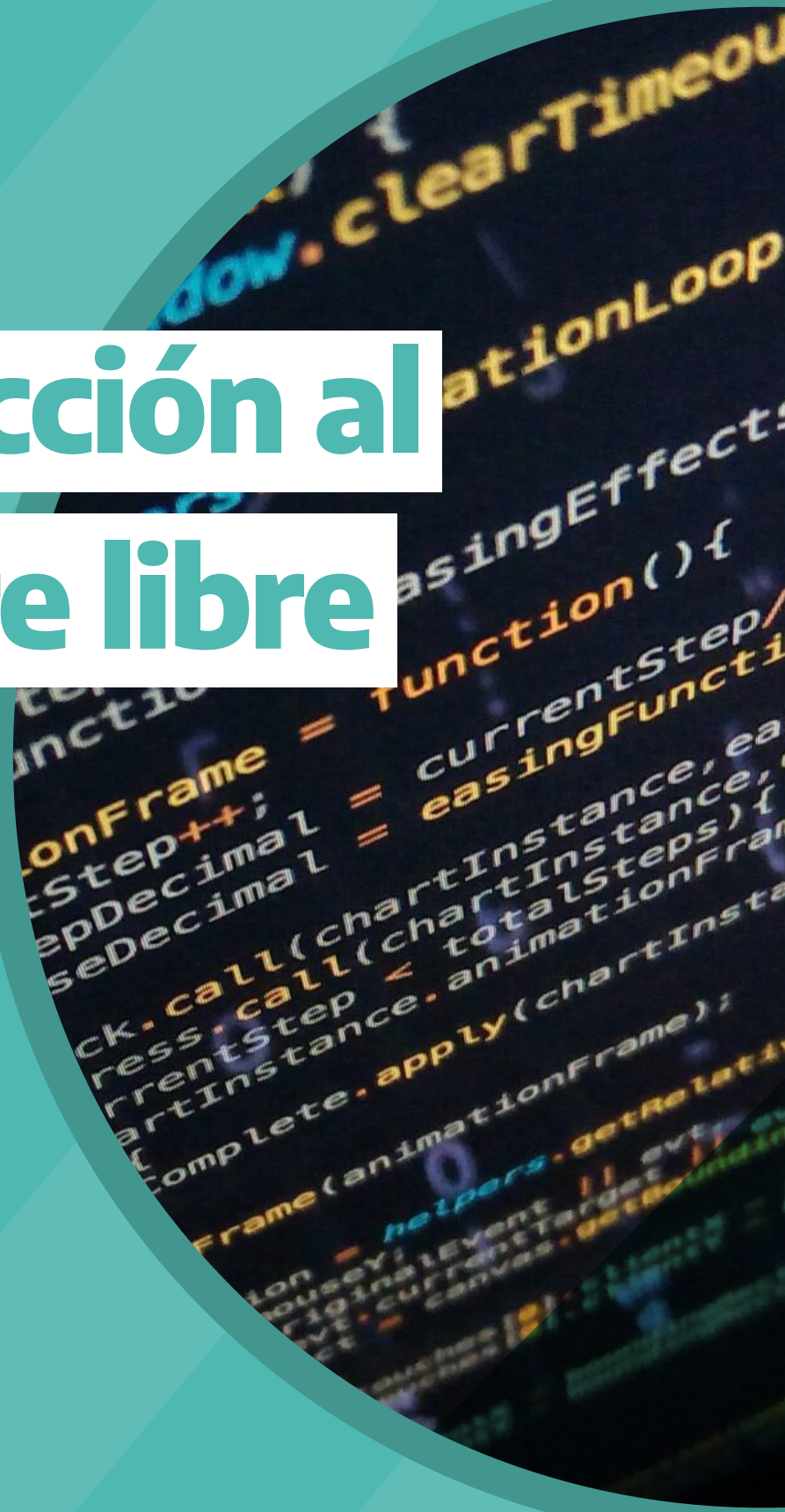


Punto  
Digital

Plataforma de  
Aprendizaje Virtual

# Introducción al software libre



Jefatura de  
Gabinete de Ministros  
Argentina

Secretaría de  
Innovación Pública

# Módulo 2

---

## 2. Consecuencias legales del modelo de software libre

---

El movimiento del software libre utiliza varios argumentos para defender su utilización, pero algunas veces olvida que todos estos argumentos tienen importantes consecuencias legales.

Para conocer estas consecuencias, y a la vez entender un poco mejor la iniciativa del software libre, consideramos adecuado comentar brevemente de qué tratan esos argumentos.

<b>Teoría de la génesis social del conocimiento</b>	El principio básico en la mayoría de las argumentaciones de las personas que defienden el movimiento del software libre es que el conocimiento como tal no pertenece a nadie, pues cualquier conocimiento proviene de unos conocimientos anteriores y es una copia en mayor o menor medida de otras ideas. La limitación de la copia, el uso o la reproducción del software no tiene sentido, pues dificulta la generación de nuevo conocimiento.
<b>Argumento tecnológico</b>	Uno de los argumentos básicos del movimiento del software libre es la defensa de la capacidad que debe tener cualquier persona usuaria de conocer, modificar y adaptar el programa a sus necesidades en cada momento.
<b>El costo</b>	<i>Software libre</i> no significa 'software gratuito', aunque a menudo el argumento de su bajo costo es utilizado por muchos de sus usuarios como determinante en la elección. La distribución con el código fuente y la autorización de copias y distribuciones posteriores ilimitadas, hace que el concepto de "pago por licencia" del software tradicional pierda su sentido.



<p><b>Mayor difusión</b></p>	<p>Quienes defienden el software libre sostienen que distribuir una obra mediante una licencia libre (en el sentido de asegurar al usuario las cuatro libertades básicas) es la mejor manera de obtener beneficios de la publicación de contenidos cuando no se es una persona con autoría consagrada que pueda obtener beneficios de un copyright cerrado.</p>
------------------------------	---

## 2.1. La libertad del software

El derecho de autor es un término jurídico que describe los derechos concedidos a quienes crean obras literarias, artísticas y científicas.

**Una obra puede definirse como la expresión formal  
de una idea o sentimiento  
que se quiere comunicar al público.**

El tipo de obras que abarca el derecho de autor incluye: obras literarias, como novelas, poemas, obras de teatro, documentos de referencia, periódicos y programas informáticos; obras artísticas; obras científicas; bases de datos; películas, composiciones musicales y coreografías; obras arquitectónicas; publicidad, mapas y dibujos técnicos.

La protección tiene incidencia sobre la forma, el continente, la expresión de la idea creativa, **pero no sobre el contenido**. Ni el material de inspiración (hechos, fechas...) ni las ideas están protegidas por los derechos de autor. Así, el concepto de obra incluye los programas informáticos, en cuanto a su código, pero no el algoritmo interactivo que representa.

### Derechos de autor

Los derechos de autor encuentran su origen en el momento histórico en que las obras se empezaron a explotar mediante la reproducción en soportes físicos (copistas medievales) y, posteriormente, a distribuirse con la aparición de la imprenta. Las editoriales, como colectivo, se beneficiaron directamente del invento de la imprenta, pues las obras se transformaron en objetos comerciales con posibilidad de proporcionarles beneficios económicos. Pero, al mismo tiempo, el uso de la imprenta dio lugar a la aparición de la primera "piratería" intelectual y presionaron a quienes legislaban para obtener protección y asegurar sus rendimientos económicos. De esta manera, el primer objetivo de los derechos de autor fue el de regular y proteger a quienes editaban contra el plagio.

La evolución de los derechos de autor a nivel nacional, y posteriormente internacional, tiene relación directa con su extensión a distintos tipos de obras y la adaptación de sus normas a las características propias de cada tipo de obra a la que iba extendiendo la protección.

### ¿Derechos de autor o copyright?

Los términos *derechos de autor* y *copyright* se utilizan frecuentemente como traducciones de un mismo concepto, pero es necesario tener en cuenta que bajo esta doble denominación se encuentran dos concepciones diferentes del derecho de autor que conviven (y a veces compiten) en el mundo actual:

- El término *author's right* –'derecho de autor' en español– está basado en la idea, nacida en Europa, de que una obra está estrechamente relacionada con quien la crea y donde el acento se pone en la persona autora. La base de los derechos de autor es, en contraposición al *copyright*, el reconocimiento de unos derechos morales irrenunciables e intransmisibles a toda persona responsable de autoría.
- El término *copyright* –'derecho de copia' en español– proviene de la tradición anglosajona y se basa en la idea de que lo más importante no es la persona autora, sino la obra y los derechos de propiedad que tienen sobre sus creaciones, con los que se puede comerciar sobre la base de los principios económicos.

Como simplificación podríamos decir que el sistema de derechos de autor es más personalista y protege la creación como extensión de la persona. En cambio, el sistema *copyright* es más colectivista y protege la creación para estimular al beneficio de interés general.

### Derechos de autor y software

Inicialmente, los programas de las computadoras no se comercializaban de manera independiente pues se distribuían con el hardware y su protección se confundía con la del conjunto del bien comercializado (fenómeno conocido como *bundling*). Además, no existían tecnologías que hicieran posible la realización de copias de los mismos o su utilización fuera del dispositivo de manera generalizada, por lo que no existía una preocupación por su protección.

La necesidad de protección empezó a plantearse a finales de los años sesenta cuando, a causa de la legislación antimonopolio norteamericana, IBM se vio obligada a separar sus negocios de hardware y software. Consecuentemente, los programas para computadoras empezaron

a estandarizarse y comercializarse en soportes móviles y la copia se generalizó.

## El sentimiento de necesidad de protección de los programas para computadoras generó un debate sobre qué figura legal era la más adecuada para protegerlos.

En ese primer momento, se entendió que podía ser objeto de protección por la figura jurídica de las patentes (el programa de computación como invención) y así lo interpretó la jurisprudencia de Estados Unidos en varias ocasiones. Sin embargo, a finales de los sesenta y principios de los setenta, quienes estaban a cargo de las legislaciones nacionales y los tratados internacionales empezaron a rechazar la protección del software como patente. En consecuencia, los países más industrializados (con gran presión de las grandes empresas informáticas) buscaron vías alternativas de protección del software:



1. Por un lado, se planteó la posibilidad de crear una protección específica o sui generis (el programa como una creación de tipo nuevo). Se pretendía establecer un mecanismo de protección con una duración de 4 a 9 años y un registro obligatorio, aunque esta iniciativa nunca llegó a prosperar.
2. Por otro lado, se planteó la posibilidad de proteger el software mediante la figura de los derechos de autor y copyright (el programa como obra literaria), y ésta fue la opción que triunfó.

En síntesis, como venimos desarrollando hasta el momento, el software está protegido por el sistema de los derechos de autor, de modo que dicho objeto legal se asimila a una obra literaria. La elección de este sistema de protección se fundamentó en las características de automatismo, armonización internacional, simplicidad y no incompatibilidad con el sistema de patentes. A nivel internacional, los principales tratados sobre la materia son:

- Convención de París para la Protección de la Propiedad Industrial de 1883.
- Convenio de Berna para la Protección de Obras Literarias y Artísticas de 1886.
- Acuerdo sobre los Aspectos de la Propiedad Intelectual Relacionados con el Comercio de 1994 (ADPIC, o TRIPS en inglés).
- Tratado de la OMPI sobre Derecho de autor de 1996.



## 2.2. Autoría en el software

La autoría en el software tradicional permite básicamente tres modalidades:

1. La **empresa-titular**, en el supuesto de software creado en el marco de una relación laboral entre la persona titular y la persona empleadora y especialistas en programación
1. La **consultoría-titular** en el supuesto de compañías que crean software por encargo.
1. Finalmente, están todos los supuestos de **autoría independiente** (obra en colaboración, colectiva, autoría individual, etc.) según el concreto modelo de desarrollo del software.

Las particularidades del modelo de desarrollo del software libre añaden dificultades a esas modalidades tradicionales mencionadas y nos obligan a realizar un análisis específico. De esta manera, existen diferentes posibilidades de autoría en el software libre:

1. Algunas aplicaciones pueden considerarse **obras en colaboración** si el concreto modelo de desarrollo del software implica que varias personas autoras contribuyan a la creación de una sola aplicación (obra) y las contribuciones de cada una puedan distinguirse (por ejemplo, con avisos de autoría en los módulos).
1. Como hemos visto, bajo ese régimen **cada persona autora es titular de su contribución, aunque la explotación de la obra se realiza conjuntamente**. Esto implica que si se opta por un modelo de desarrollo del software de este tipo, es esencial acordar entre quienes están a cargo de la autoría la licencia que se aplicará al software y el régimen de explotación de la obra resultante.
1. Otras aplicaciones de software serán **obras colectivas si no pueden diferenciarse las contribuciones de los diferentes programadores**. Ejemplos de aplicaciones de software libre que podrían considerarse como obras colectivas (si fueran desarrolladas y diseminadas bajo el sistema de derechos de autor) incluyen el programa Mozilla, que es coordinado por Mozilla.org, o las aplicaciones de la Free Software Foundation.
1. Para prevenir posibles problemas futuros, los proyectos de desarrollo de software libre que opten por esta opción deben **asegurarse de que cada persona autora-contribuyente transfiere por escrito todos los derechos a la entidad coordinadora** para que ésta pueda administrar correctamente los derechos de propiedad intelectual de la aplicación, en concreto, determinar el régimen de licencia, asegurar la defensa de cualquier infracción, etc.

En la creación del software libre, es usual la aportación de diferentes especialistas en programación. Para saber quién es el titular de la autoría, quién ostenta los derechos sobre la obra y quién está legitimado para ejercitar acción por copia, es necesario delimitar si el resultado es una obra colectiva o una obra en colaboración. La respuesta dependerá del concreto modelo de desarrollo utilizado, de la existencia o una persona que coordine, de la separabilidad o no de las diferentes aportaciones y por supuesto de los pactos concretos establecidos.

### Obra original y obra derivada



**Una obra original es aquella obra que constituye una creación autónoma e independiente, aunque se publique conjuntamente con otras.**

**Una obra derivada, en cambio, es aquella obra nueva que incorpora una obra preexistente, sin la colaboración de la persona responsables de esta última, aunque sí con su autorización (para ser obra derivada lícita).**

Existirán, por lo tanto, dos obras diferenciadas y con dos titularidades: la obra original y la obra derivada. Aquella obra que incorpora una obra preexistente pero sin autorización (expresa o implícita a través de la colaboración de la persona autora) produce una infracción de los derechos de autor del titular de la obra original. En el software libre, éste es un tema polémico, por las dificultades de realizar una diferenciación entre obra derivada, obra en colaboración y obra colectiva. Las dificultades se plantean básicamente en dos entornos:

1. En relación con el sistema de creación del software libre, esta diferenciación es muy relevante, pues como hemos visto, él mismo se basa en la incorporación, agregación y modificación de obras preexistentes. El modelo usual de desarrollo de software libre implica una cantidad importante de contribuyentes a la creación de una misma aplicación. Algunas personas autoras aportan código nuevo (por lo tanto, crean obras en colaboración o colectivas), otras corrigen o mejoran un código existente (por lo tanto, creando una obra derivada). Asimismo, pueden agregar un aviso de autoría o no.
2. En cuanto al uso y la explotación del software libre, esta diferenciación entre obra original y derivada también es importante. La mayoría de las licencias de uso permiten la modificación (adaptación, traducción, etc.) de las aplicaciones y, por lo tanto, la creación de obras



derivadas. Algunas licencias de software libre controlan la difusión y el uso de estas obras derivadas (por ejemplo, la GPL), mientras que otras no (la BSD).

En definitiva, es importante entender bien la definición de obra derivada para saber si un desarrollo realizado sobre la base de una aplicación libre (por ejemplo, integrándola, modificándola, usándola, etc.) puede considerarse:

- Una obra derivada de la original.
- Una obra original que utilice la misma pero no la modifique.





Jefatura de  
Gabinete de Ministros  
Argentina

Secretaría de  
Innovación Pública