

CIENCIA ABIERTA UGR

mcogolludo@ugr.es

@Makova_

Software Libre (Free Software)

FREE quiere decir “Libre”, no “Gratis”.

(“Libre” como el sol cuando amanece, no como en “barra libre”).

FSF (Free Software Foundation)



Richard Stallmann

Libertades del Software

- 0: Usar el programa
- 1: Estudiarlo y adaptarlo
- 2: Distribuir copias
- 3: Hacer mejoras y publicarlas

Libertad para usar el programa

**Sin restricciones por objetivo, modo de
uso, ubicación, etc.**

Libertad para estudiar el programa

**...Y para adaptarlo a las propias
necesidades.**

Necesita acceso al código fuente.

Libertad para distribuir copias

Gratis, o por un precio.

Libertad para mejorar el programa

...Y para compartir esas mejoras

OSI (Open Source Initiative)



Eric S. Raymond

Criterios de la OSI (Open Source Definition)

1. Redistributions must include the source code.
2. You must be able to change the software or programs that are distributed in order to adapt them to your needs; you must also be able to redistribute the changed versions to others without permission from the original author or distributor.
3. You may not discriminate against anyone for doing anything with the software.
4. You may not restrict anyone's right to redistribute or further develop the source code.
5. You may not limit how users may use the software or how it may be redistributed beyond what is required by the license.
6. You may not limit users' freedom to run the software and to redistribute it along with their other software.
7. You may not limit how users may study, change, and improve the software.
8. You may not limit users' freedom to redistribute the software along with their other software.
9. You may not limit how users may study, change, and improve the software.
10. You may not limit how users may redistribute the software along with their other software.

1 Redistribución libre

La licencia no debe impedir la venta o distribución del programa, ni exigir pagos o regalías por su venta.

2 Código fuente

El programa debe incluir su código fuente o disponer del modo de acceso a este, y la licencia debe permitir su distribución.

3 Obras derivadas

La licencia debe permitir la realización y distribución de modificaciones y obras derivadas.

4 Integridad del código del autor

Si se ha modificado el código fuente, se puede impedir su distribución si se provee de

5 No discriminación de personas

... O grupos.

6 No discriminación de actividad

No se pueden discriminar sectores o ámbitos de actividad como, por ejemplo “Fabricación de zapatos”, “Investigación genética” o “Difusión de literatura religiosa”.

7 Distribución de la licencia

Los derechos del programa se cederán a todos los que los reciban, sin obligar a firmar ninguna licencia adicional.

8 Independencia de un producto

Los derechos del programa no deben depender de que este forme parte de un producto, paquete o versión de software.

9 No limitar otro software

La licencia no debe imponer restricciones sobre otro software que se distribuya conjuntamente con este.

10 Neutral respecto a la tecnología

Ninguna disposición de la licencia puede depender de una tecnología o tipo de interfaz determinado.

Software Privativo

Es lo contrario del Software Libre o de Fuentes Abiertas.

Es aquel que le falta alguna de las libertades de la FSF y/o no cumple alguno de los criterios de la OSI

Software Privativo

No son Software Libre

Freeware
Shareware

Versiones “demo” limitadas
Código fuente (soloamente) disponible
Cualquier licencia que prohíba la
redistribución

Copyleft



Las cláusulas Copyleft exigen que el software derivado se distribuya con la misma licencia y en las mismas condiciones que el original.

No se puede crear software privativo a partir de software libre que tenga Copyleft

Copyleft

El Copyleft es “vírico”.

Copyleft

Copyleft NO es lo opuesto a Copyright

Es una extensión del Copyright

Copyleft

Copyleft NO es lo opuesto a Copyright

Es una extensión del Copyright

Se aplica en la distribución, pero no obliga a distribuir el software.

Licencias

Una licencia es una cesión de derechos por parte del propietario del Copyright hacia el licenciatario.

Licencias

Toda licencia tiene dos caras:

Licencias

Aseguran los
derechos del
usuario

*Detallando las
autorizaciones
para el uso del
software*

Licencias

Aseguran los derechos del usuario

Protege los derechos del titular

Detallando las autorizaciones para el uso del software

Al especificar los derechos no cedidos y las condiciones que debe cumplir el usuario

Licencias

Una licencia es una cesión de derechos por parte del propietario del Copyright hacia el licenciatario.

Licencias (De Software Libre)

Una licencia es una cesión de derechos por parte del propietario del Copyright hacia el licenciatario.

...Y es libre si cumple los mencionados requisitos de la FSF y la OSI

Licencias (De Software Libre)

Existen MONTONES de licencias de Software Libre (La OSI ha certificado unas 70).

Licencias (De Software Libre)

Existen MONTONES de licencias de Software Libre (La OSI ha certificado unas 70).

Pero no todas son iguales.

Licencias (De Software Libre)

La principal diferencia entre unas y otras está en las condiciones que imponen a la redistribución del software.

Licencias (De Software Libre)

La principal diferencia entre unas y otras está en las condiciones que imponen a la redistribución del software.

Concretamente, en su grado de Copyleft.

Licencias (De Software Libre)

Copyleft “**fuerte**”

Son aquellas que exigen que cualquier modificación o trabajo derivado que se distribuya debe tener la misma licencia que el original.

Licencias (De Software Libre)

Copyleft “**fuerte**”

GPL
Affero
CPL
Sleepycat

Licencias (De Software Libre)

Copyleft “suave”

Son las que permiten integrar el software bajo otra licencia, pero la parte original debe conservar su licencia original (normalmente, se usan para librerías o módulos)

Licencias (De Software Libre)

Copyleft “*suave*”

LGPL
MPL
OSL

Licencias (De Software Libre)

Sin Copyleft (“permisivas”)

Las que permiten redistribuir el código o sus modificaciones y derivados bajo cualquier otra licencia (incluso privativas)

Licencias (De Software Libre)

Sin Copyleft (“permisivas”)

BSD
MIT
Apache
ASL

Licencias (De Software Libre)

Además, cada licencia concreta, al igual que en el Software privativo, cubre los distintos aspectos que sus respectivos autores han considerado necesarios.

(Protección de marcas registradas, modo de distribuir las modificaciones, exención de responsabilidad, jurisdicción...)

Licencias (De Software Libre)

La cesión de derechos bajo licencia libre
¿es una renuncia?

Licencias (De Software Libre)

La cesión de derechos bajo licencia libre
¿es una renuncia?

NUNCA

Licencias (De Software Libre)

¿Las condiciones establecidas por las licencias libres son válidas?

Licencias (De Software Libre)

¿Las condiciones establecidas por las licencias libres son válidas?

Al igual que con el Software privativo, siempre que no existan cláusulas particulares que puedan considerarse nulas desde el punto de vista jurídico, como condiciones abusivas hacia el licenciatario.

Licencias (De Software Libre)

¿Cómo hacer cumplir las obligaciones de la licencia?

El propietario del Copyrighth mantiene todos los derechos legales sobre el software.

Vayamos a lo práctico

Vale. Muy bonito todo esto pero...

Vayamos a lo práctico

Vale. Muy bonito todo esto pero...

¿Cómo decido qué licencia usar?

Vayamos a lo práctico

Algunos consejos:

No te compliques: Si no vas a redistribuir el software, no necesitas licencias.

Vayamos a lo práctico

Algunos consejos:

No te compliques: Si no vas a redistribuir el software, no necesitas licencias.

No reinventes la rueda: Usa una de las licencias reconocidas.

Vayamos a lo práctico

Algunos consejos:

No te compliques: Si no vas a redistribuir el software, no necesitas licencias.

No reinventes la rueda: Usa una de las licencias reconocidas.

Pero, sobre todo, ten claro para qué quieras la licencia...

Vayamos a lo práctico

¿Es por motivos éticos?

¿Voy a dar soluciones comerciales basadas en este software?

¿Parto de un código que ya tiene una licencia?

¿El software es parte de un programa universitario o el resultado de una investigación?

¿Es parte de una iniciativa pública?

Vayamos a lo práctico

Dependiendo de las circunstancias concretas de cada caso, será más adecuada una u otra.

Vayamos a lo práctico

Dependiendo de las circunstancias concretas de cada caso, será más adecuada una u otra.

Y en **Granada Hacklab** te podemos echar una mano...

Licencias Duales

El propietario del Copyright puede distribuirlo según las licencias que prefiera.

Incluso con varias distintas dependiendo del cliente o las circunstancias.

Licencias Duales

Un Software, dos licencias:

Una licencia libre para usuarios libres, y
una licencia privativa para usuarios
privativos.

Licencias Duales

Dos programas, dos licencias:

Una versión libre para usuarios libres, y
una versión privativa con funcionalidades
“extra”.

Licencias Duales

Un programa, una licencia (con opción a cambiar):

Algunas licencias “menos restrictivas” permiten su cambio estratégico (LGPL → GPL).

Compatibilidad entre Licencias

Por definición, es posible integrar software libre de distintas fuentes para obtener un tercer producto, incluso aunque sus licencias sean distintas.

Pero, si se quiere distribuir este producto, las licencias deben ser compatibles.

Compatibilidad entre Licencias

Decimos que dos licencias son compatibles cuando es posible **integrar el código de un programa en otro bajo una licencia distinta sin que la redistribución de la obra resultante infrinja la licencia del primero.**

Compatibilidad entre Licencias

O sea:

No puedo hacer cosas como coger código
bajo una licencia con Copyleft y
distribuirlo con una licencia sin Copyleft.

Estaría incumpliendo la licencia original.

Compatibilidad entre Licencias

O sea:

No puedo hacer cosas como coger código bajo una licencia con Copyleft y distribuirlo con una licencia sin Copyleft.

Pero sí puedo hacerlo al revés.
(Hacer un trabajo Copyleft de un original sin él)

Proyectos de Software Libre

Proyecto básico: se libera un software de nueva creación.

Proyecto complejo: se integran componentes libres de terceros.

Proyecto maduro: se ha publicado el código y la comunidad empieza a contribuir.

Proyectos de Software Libre

Para todos ellos, debemos asegurarnos jurídicamente, cumpliendo una serie de objetivos:

Proyectos de Software Libre

Asegurar la titularidad de los derechos para poder liberar el software.

Seleccionar una licencia de distribución que asegure los objetivos del proyecto.

Cumplir con las obligaciones establecidas en las licencias de los componentes integrados (y evitar su incompatibilidad)

Proteger la reputación del proyecto y la calidad de su software.

Proyecto Básico

Desarrollo:

Garantizar la titularidad original del software.

Evitar los riesgos de patentes.

Delimitar el perímetro de confidencialidad.

Distribución:

Selección de la licencia.

Selección y registro de una marca.

Distribución en línea.

Proyecto Complejo

Desarrollo:

Identificación de la licencia entrante.

Determinar su compatibilidad con la licencia de saliente.

¿Existen patentes sobre componentes?

Distribución:

Cumplimiento de las condiciones de las licencias sobre componentes.

Uso de marcas de componentes en la documentación y publicidad

Con contribuciones externas

Titularidad y cesión de derechos de las contribuciones al proyecto:

Averiguación del origen.

Cesión completa o licencia de uso bajo la licencia del proyecto / licencia permisiva.

Acuerdo sobre contribuciones.

Resumiendo...

El objetivo de todo esto es asegurarse de que se cumplen las condiciones de las respectivas licencias, sin vulnerar derechos de terceros.

Modelos de negocio

O lo que es lo mismo:

Modelos de negocio

O lo que es lo mismo:

¿Donde está la pasta?

Modelos de negocio

El negocio del Software Libre no puede basarse en la posesión de un monopolio sobre el producto.

Modelos de negocio

El negocio del Software Libre no puede basarse en la posesión de un monopolio sobre el producto.

Por tanto, la sola venta del software en sí difícilmente será un buen negocio.

Modelos de negocio

El negocio del Software Libre no puede basarse en la posesión de un monopolio sobre el producto.

Por tanto, la sola venta del software en sí difícilmente será un buen negocio.

Hay que dar un Valor Añadido

Modelos de negocio

Proyectos originales que prestan servicios de implantación y soporte sobre sus propios productos.

Empresas independientes que integran componentes libres para crear paquetes.

Diseño de soluciones complejas a medida para los clientes.

Prestación de servicios de soporte, mantenimiento y formación.

Modelos de negocio

Proyectos originales que prestan servicios de implantación y soporte sobre sus propios productos.

Empresas que crean software, y asisten a sus clientes en la instalación e implantación y en tareas de soporte a usuario y/o programación de soluciones a medida.

Modelos de negocio

Empresas independientes que integran componentes libres para crear paquetes.

Integrando varias aplicaciones de Software Libre programado por terceros en productos complejos.

Modelos de negocio

Diseño de soluciones complejas a medida para los clientes.

Adaptación de uno o varios productos de Software Libre a la medida de las necesidades del cliente.

Modelos de negocio

**Prestación de servicios de soporte,
asesoría, mantenimiento y
formación.**

Empresas especializadas en aportar servicios de valor añadido sobre el Software Libre de terceros.

Protección Jurídica del Software

El Software Libre cuenta con los mismos mecanismos de protección jurídica que el Software Privativo.

Protección Jurídica del Software

El código literal del programa y su documentación están protegidos expresamente por los **derechos de autor**.

(Automático, gratuito e internacional)

Protección Jurídica del Software

Las funcionalidades y procesos de un programa pueden protegerse (aunque de manera muy controvertida) por la figura de **la patente**.

El Software no puede patentarse en Europa.

Protección Jurídica del Software

El **derecho de marcas** concede un derecho exclusivo y excluyente sobre el uso de una marca relativa a un software.

Protección Jurídica del Software

El **secreto industrial** permite proteger la información confidencial concerniente a la creación del software, su diseño o sus modelos de implementación y de negocio que puedan tener un alto valor comercial.

Protección Jurídica del Software

El **derecho de marcas** concede un derecho exclusivo y excluyente sobre el uso de una marca relativa a un software.

CREATIVE COMMONS



creative
commons

DERECHOS DE AUTOR

- Los tienes por el solo hecho de crear
- Se divide en:
 - Morales
 - Patrimoniales

Todo uso requiere permiso expreso

- Excepciones (muy excepcionales):
 - Copia privada
 - Derecho de cita
 -y poco más.

Problema:

- Las leyes de propiedad intelectual están pensadas para “proTEGER” la obra contra su uso antes que para permitirlo.

El copyright se queda corto

- La gente esta ideando nuevos usos creativos
 - Subtítulos
 - Traducciones
 - Fanfics, Doujins
 - Compartir la propia obra
 - Mezclas, samples y collages

CULTURA LIBRE

Corriente de pensamiento que promueve la libertad en la distribución y modificación de trabajos creativos basándose en el principio del contenido libre para distribuir o modificar trabajos y obras creativas

https://es.wikipedia.org/wiki/Cultura_libre

Obra cultural libre

- Es aquella que permite:
 - Usar y ejecutar la obra
 - Estudiar la obra y aplicar la información
 - Redistribuir copias
 - Distribuir obras derivadas

En ROJO las que NO son libres

Es decir, que no cumplen los criterios
para ser
“Obra cultural libre”

Símbolos de CC



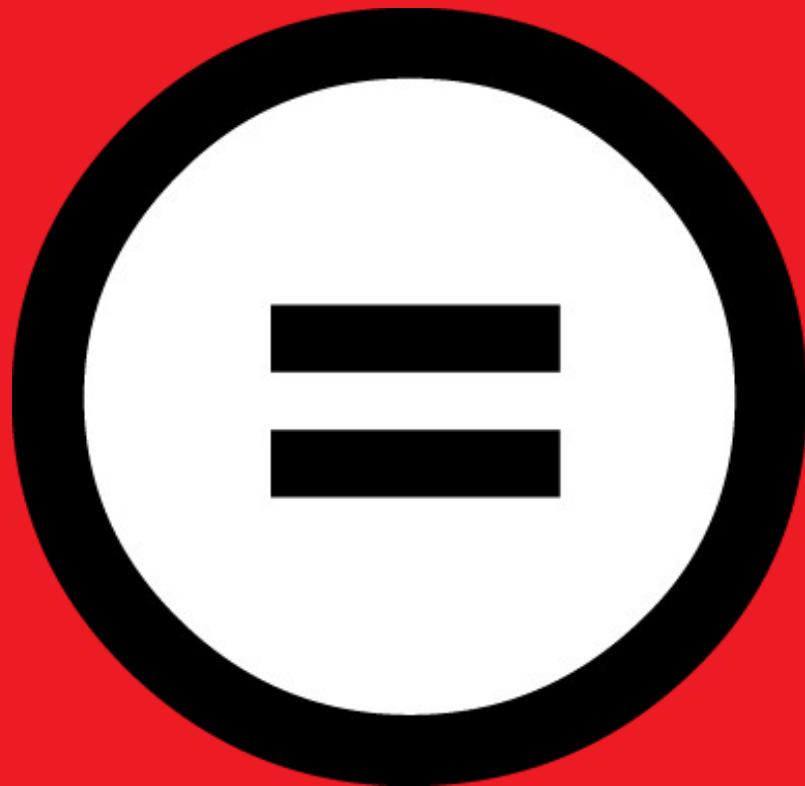
Reconocimiento (Attribution)



No comercial (Non commercial)



**Sin obras derivadas
(No Derivative Works)**



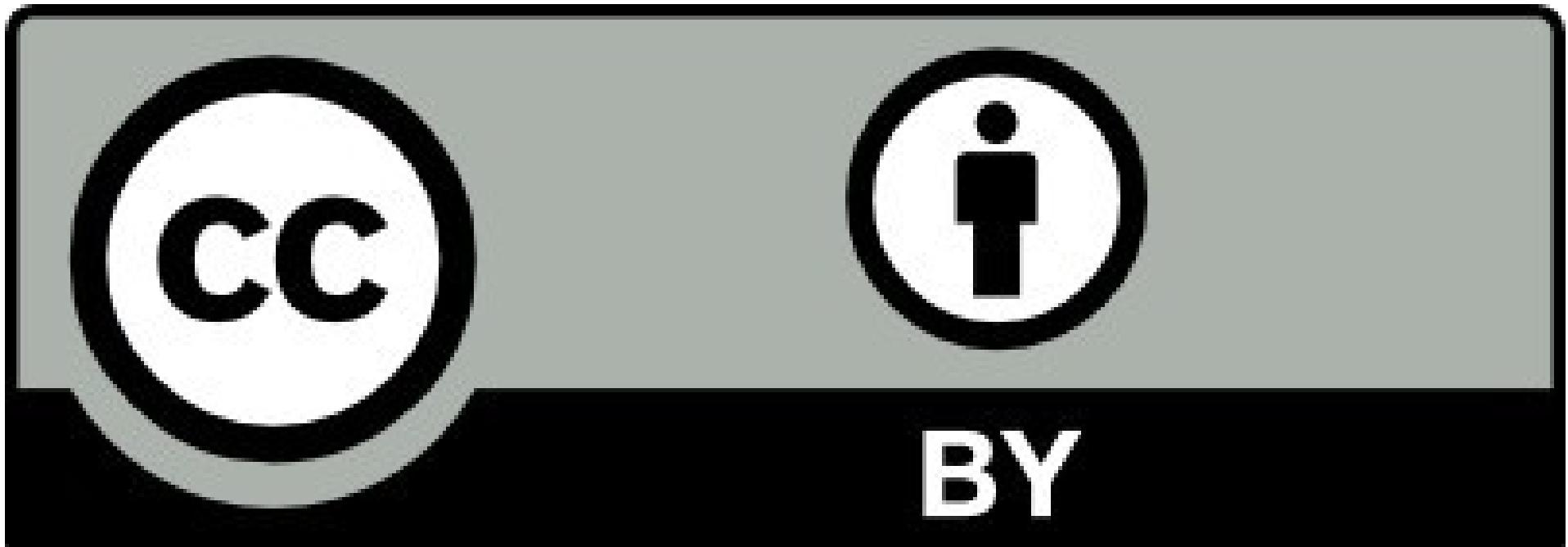
Compartir igual (Share alike)



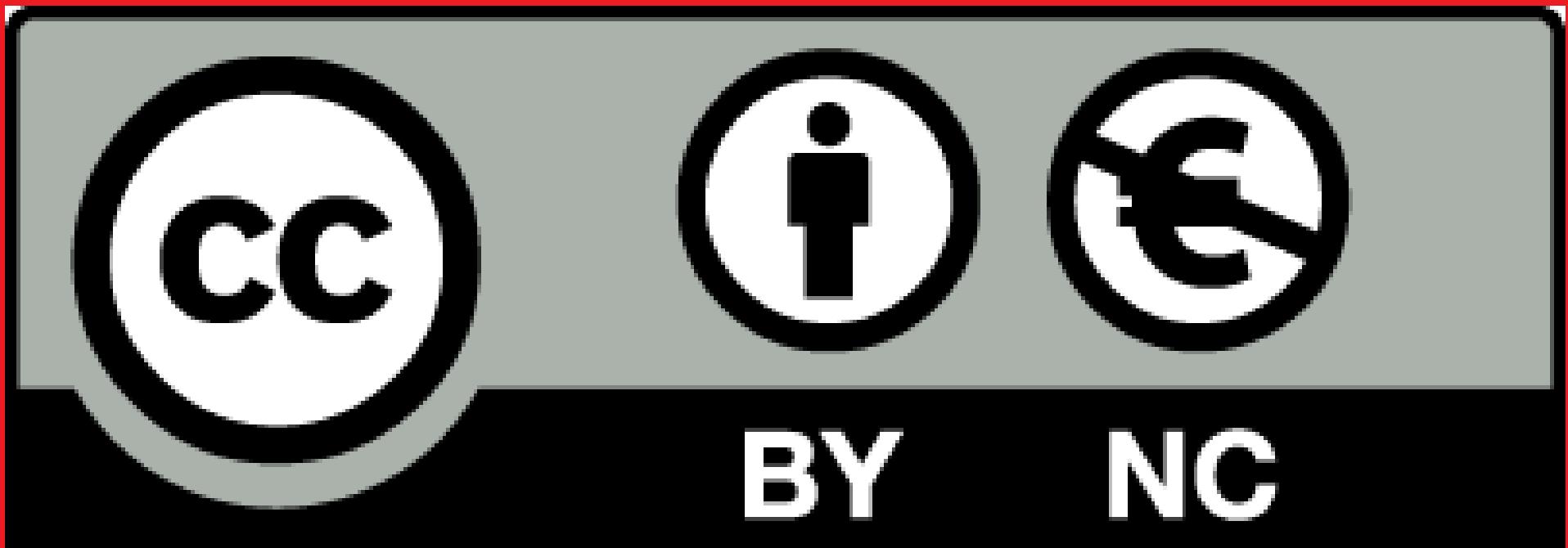
Ahora, veamoslo en acción

- Todo junto

Reconocimiento (by)



Reconocimiento – NoComercial (by-nc)



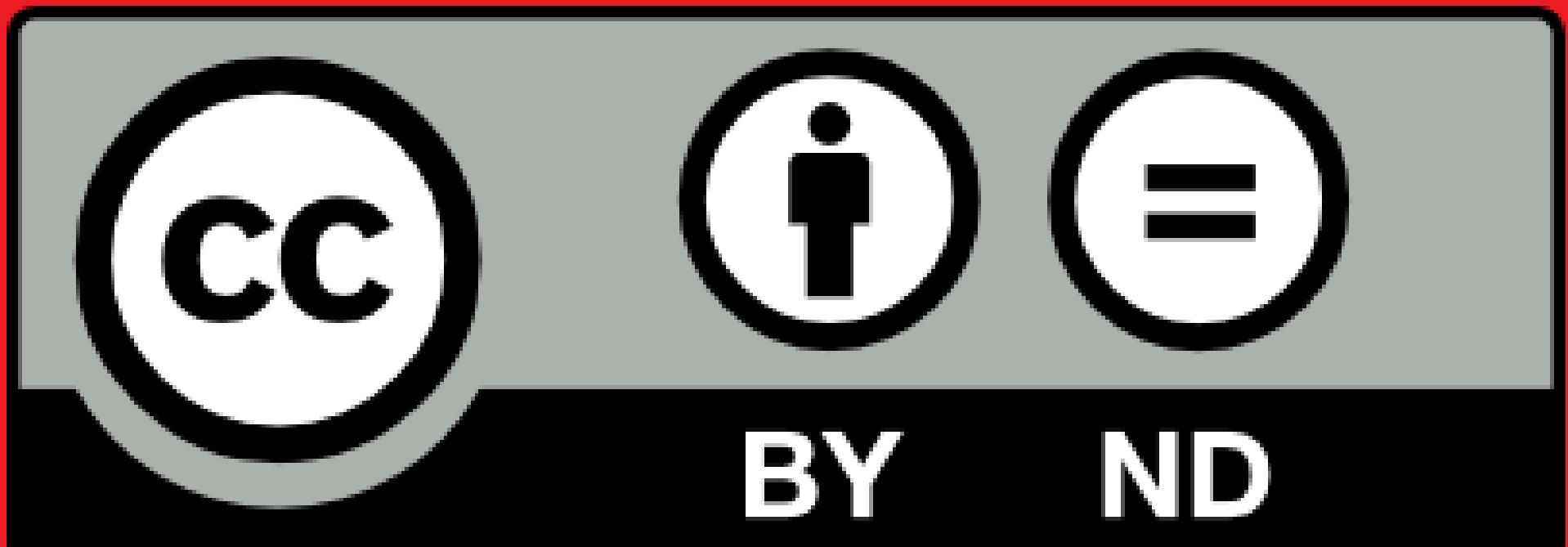
Reconocimiento – NoComercial Compartirlgual (by-nc-sa)



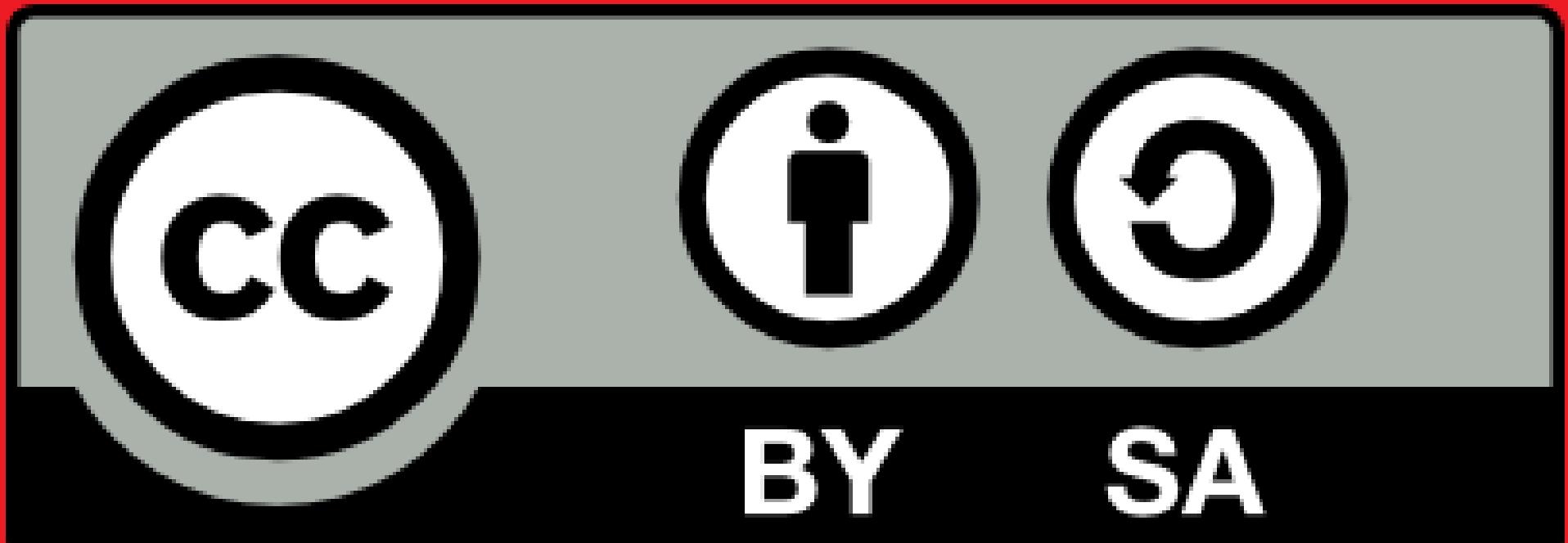
Reconocimiento – NoComercial - SinObraDerivada (by-nc-nd)



Reconocimiento SinObraDerivada (by-nd)



Reconocimiento – CompartirIgual (by-sa)



**Obra original por
Angel Pablo Hinojosa
© 2009**



Creada para la Oficina de
Software Libre UGR

Y modificada por Manu Cogolludo de
Granada Hacklab, para el curso de
CIENCIA ABIERTA UGR