

EL SOFTWARE LIBRE COMO HERRAMIENTA PARA LA AUTOGESTIÓN Y COMO PROYECTO DE AUTOGESTIÓN EN SÍ MISMO.

Por **Germán Dartsch**.

Cuando me invitaron a dar esta charla, yo llevaba un tiempo estudiando el software libre, pero desde otros puntos de vista, no me había puesto a ver de cerca el tema de la autogestión y cómo puede ser el software libre una herramienta para autogestionarse. Así, me puse a hacer una investigación rápida, y llegué a dar con una definición de lo que significa autogestión. Se las leo:

Definición de Autogestión.

En economía política y sociología, autogestión puede referirse a la economía autodirigida, un sistema económico de naturaleza socialista, basado en formas diversas, mecanismos y métodos colaborativos y descentralizados, que incluyen un proceso de toma de decisiones autónomo dentro de las empresas. Los postulados de la autogestión son:

- la supresión de distinciones entre dirigentes y dirigidos,
- la transparencia y la legitimidad de las decisiones,
- la no apropiación por algunos de la riqueza generada por la colectividad,
- la afirmación de la capacidad de los dirigidos para organizarse sin dirigentes.

Esta concepción es construida explícitamente en contra de las prácticas jerárquicas, autoritarias y verticales de organización. Es en cierta manera una forma de organización colectiva.

En base a esta definición me pude dar una idea de cómo puede usarse el software libre como herramienta para la autogestión, pero también me dí cuenta de hasta qué punto el software libre es, en sí mismo, un proyecto (o proyectos) de autogestión.

Es por esto que decidí armar mi exposición en base a la forma en que el software libre cumple con estos cuatro supuestos que la definición que les acabo de leer propone como base de la autogestión.

Pero antes, ¿Qué es el software libre?

El software libre, puede definirse de dos maneras, como producto y como movimiento.

Como producto es una gama de programas informáticos que cumplen con cuatro condiciones, llamadas por sus partidarios las libertades del software libre:

Libertades del Software Libre.

Libertad	Descripción
0	la libertad de usar el programa, con cualquier propósito.
1	la libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a tus necesidades.
2	la libertad de distribuir copias del programa, con lo cual puedes ayudar a tu prójimo.
3	la libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.
Las libertades 1 y 3 requieren acceso al código fuente porque estudiar y modificar software sin su código fuente es muy poco viable.	

¿Y qué vendría a ser código fuente?

Es un lenguaje intermedio entre el programador y la máquina. La máquina entiende instrucciones en código binario, en ceros y unos, un lenguaje que difícilmente pueda entender y menos manejar un ser humano. El código fuente es, en cambio, un lenguaje que los programadores pueden llegar a dominar y es lo que utilizan para programar.

El acceso al código fuente es la base de todo el software libre, puesto que el código fuente es lo que permite entender cómo funciona un programa, el poder investigarlo y el poder modificarlo y trabajar sobre él. El software propietario, en cambio, se basa en el ocultamiento del código fuente, para evitar justamente estas libertades y que el programa sea monopolio exclusivo de sus dueños.

Ahora bien, **como movimiento social** el software libre puede definirse como el movimiento que surge en los años 80 de los institutos de investigación de software y que, acostumbrados a investigar software y compartirlo con libertad, reaccionan contra la privatización del software una vez que éste se empieza a convertir más en un negocio y menos en un objeto de estudio.

Este movimiento es el que produce el software libre y lo justifica desde puntos de vista que ponen énfasis o en la superioridad técnica del software con código fuente libre o en la pérdida de libertad que sufre un usuario de software cuando el código fuente es ocultado. Sus reivindicaciones los llevaron a cuestionar las leyes de copyright y a acuñar (o perfeccionar) su propia modalidad de trabajo: el **modelo bazar**, un modo de trabajo descentralizado en el que cada miembro de la comunidad puede aportar a distintos proyectos de software lo que le parezca adecuado o modificar lo que existe sin restricciones y con horizontalidad. Suena mucho a autogestión.

Ahora, podemos interpretar de distintas maneras el movimiento del software libre. O bien, podemos decir que sus reivindicaciones llevan a una defensa del liberalismo extremo por el énfasis que se pone en el dejar hacer, dejar pasar, y dónde ya no es el afán de lucro lo que lleva a participar del modelo, sino el afán de ego, como dicen algunos autores, específicamente Eric Raymond.

O desde una posición más de izquierda, puesto que la liberación del código fuente puede suponer dos cosas:

1. La desenmascaración del fetichismo de la mercancía, al dejar a la vista el cómo se hace un producto de software y quiénes participan en su elaboración.
2. Podríamos suponer que liberar el código fuente, que es un medio de producción clave para la industria del software, es una reivindicación de la socialización de los medios de producción del software.

Pero estos temas podrían dar lugar a muchos debates y nos iríamos al tema. Por tanto, vamos al centro de la exposición:

¿Cómo satisface el software libre cada uno de los 4 postulados de la autogestión?

Vayamos uno por uno:

1. La supresión de distinciones entre dirigentes y dirigidos:

El modelo bazar es horizontal, descentralizado y hasta anárquico a su manera (y también, organizado a su manera). No existen jerarquías a su interior, y eso permite la participación por igual de todo aquél que tenga algo que aportar. El código fuente libre refuerza la participación, pues permite que cada quién modifique un programa informático sin

autorización de nadie, y sus aportes son juzgados sólo por la comunidad en su conjunto.

2. La transparencia y la legitimidad de las decisiones:

A veces, cuando usamos por ejemplo windows, nos sale de repente algún cartel del tipo "error desconocido" o similares. En linux, cuando hay un error el cartel nos ofrece una explicación de lo que pasó, que aunque probablemente sea algo que sólo un informático entienda, se puede consultar por foros con esa info en mano, y se puede arreglar sin necesidad de acudir al "servicio técnico oficial". El software libre es transparente. El código fuente libre garantiza esa transparencia. Por eso, todos podemos ver cuando alguien hace un cambio en un programa informático libre, porque el código está a la vista de todos. En el software libre no existen las decisiones secretas o encubiertas y eso garantiza su legitimidad. De igual manera se espera legitimidad en las decisiones de un colectivo autogestionado: sin mentiras, con todo a la vista de la comunidad.

3. La no apropiación por algunos de la riqueza generada por todos:

Éste fue un punto clave en la historia del software libre. Al dejar sus programas libres, cualquiera los podría retomar y licenciarlos bajo una licencia restrictiva, matando al propósito original del software libre. Esto motivó a que hoy los que son partidarios del software libre sean también opositores al copyright, al modelo actual de propiedad intelectual y derechos de autor.

De su lucha contra el copyright surgirá la herramienta que permitirá que la creación colectiva no sea apropiada por pocas manos: el copyleft.

Copyleft es una forma de usar el copyright pero invirtiendo sus propósitos: en lugar de estar todos los derechos reservados al autor, el autor elige ceder algunos derechos, los consistentes con las cuatro libertades del software libre. Pero además, el copyleft va un paso más allá.

Todavía existe la posibilidad de que alguien tome un programa libre, lo modifique mínimamente y cree un programa propietario a partir de software libre. Y acá es donde el copyleft se luce: toda obra derivada de un producto licenciado bajo una licencia copyleft debe, obligatoriamente, llevar licencia copyleft también. Así, se garantiza que la creación

colectiva siga siendo de la colectividad.

Del concepto de copyleft surgen todos los instrumentos legales que hoy garantizan y protegen a la cultura libre como las licencias que ofrece la organización Creative Commons.

Y último supuesto.

4. La afirmación de la capacidad de los dirigidos para organizarse sin dirigentes:

Ésta es la esencia del modelo bazar. En el modelo bazar todos somos iguales, todos podemos participar. Existen, sí, líderes, pero lo son por su propia capacidad de liderazgo y no por su jerarquía, y sus seguidores los siguen por voluntad propia, porque creen en él. Es el caso de Linus Torvalds, creador de linux, o del mismo motor del movimiento del software libre: Richard Stallman. Y digo más: los líderes del modelo bazar son transitorios y son a veces más bien consejeros o guías, pero no detentan ni monopolizan el poder, el poder lo tiene la comunidad y lo delega, si lo necesita, en quien necesite según los objetivos que en común se haya planteado. El modelo bazar es democracia más que anarquía.

Así que, en resumen:

El software libre es en sí mismo un proyecto de autogestión y como tal cumple con los postulados básicos de lo que significa autogestionarse.

Es por eso que, como herramienta informática, es el mejor medio para cualquier proyecto de autogestión, porque el software libre es garantía de:

- Horizontalidad.
- Transparencia y legitimidad.
- No apropiación de la creación colectiva.
- Afirmación de la autoorganización.

Y son esos los valores básicos que sustentan la cultura libre y la autogestión.