

Políticas Públicas para el uso de Software Libre

El Software Libre son todos los programas de computadoras que se pueden compartir, estudiar, adaptar y crear versiones modificadas. La Free Software Foundation define al software libre como cualquier programa informático que respete las siguientes libertades¹:

- *La libertad de usar el programa, con cualquier propósito (libertad 0).*
- *La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.*
- *La libertad de distribuir copias, con lo que puedes ayudar a tu vecino (libertad 2).*
- *La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.*

El código fuente² de una aplicación son las instrucciones escritas en lenguaje humano que entienden las personas. Estas instrucciones luego son convertidas a lenguaje de máquina para que los entiendan las computadoras. El software privativo se distribuye la mayoría de ocasiones sin el código fuente por lo que es imposible modificarlo y saber como funciona.

Cada día son más los gobiernos del mundo que tienen políticas que favorecen la utilización del software libre. Solo en América Latina países como Brasil, Venezuela, Ecuador y Paraguay apuntan a la utilización de software libre en los gobiernos. A nivel mundial gobiernos como el de Sudáfrica³, algunos estados en la India⁴, gobiernos regionales de España como la junta de Extremadura⁵ y Andalucía,⁶ y varias otras administraciones públicas al-rededor del mundo

En el caso de Ecuador el Presidente Rafael Correa en un vídeo grabado para 18 países de América Latina como conmemoración del Festival Latinoamericano de Instalación de Software Libre 2007 (FLISOL) explica algunas razones para utilizar software libre en los estados⁷.

Por eso es necesario que todos adoptemos, tanto a nivel público cuanto a nivel privado, el software libre. De esa manera garantizaremos la soberanía de nuestros estados, dependeremos de nuestras propias fuerzas, no de fuerzas externas a la región; seremos productores de tecnología, no simples consumidores; seremos dueños de los códigos fuentes; y podemos desarrollar muchos productos que, incluso, con una adecuada articulación de nuestros esfuerzos, pueden ser de suma utilidad para las empresas públicas y privadas de la región.

1 Software Libre en la Administración Pública

1.1 Soberanía Tecnológica

Las soluciones de software privativo atan el soporte de la aplicación a la empresa que la desarrolló. Esto genera dependencia del proveedor ya que el estado no tiene el control de los sistemas sino la empresa que tiene el código fuente y los derechos para modificarlo. Un estado soberano debe

1 <http://www.gnu.org/philosophy/free-sw.es.html>

2 http://es.wikipedia.org/wiki/Codigo_fuente

3 <http://www.tectonic.co.za/wordpress/?p=1377>

4 <http://swatantryam.blogspot.com/2007/08/story-of-free-software-in-kerala-india.html>

5 <http://www.hispalinux.es/node/596>

6 <http://sia.juntaex.es/pdfs/doe/2008/18900/08040215.pdf>

7 <http://www.presidencia.gov.ec/noticias.asp?noid=9267&hl=true>

depender de sus propias fuerzas y el software libre nos da estas oportunidades.

No es suficiente utilizar software libre para tener soberanía tecnológica. Se debe llegar a la capacidad de poder modificar las aplicaciones y adaptarlas a nuestras necesidades. Así mismo se debe desarrollar software libre de manera local para no depender de fuerzas externas.

1.2 Seguridad y Privacidad

El software que utiliza un gobierno maneja información sobre los ciudadanos. Datos como la cédula de identidad, las aportaciones al seguro social, el pago de impuestos, el conteo de votos en una elección, entre otra información sensible que se la debe manejar con total transparencia.

Es importante para los ciudadanos que nuestra información se la maneje de manera transparente publicando el código fuente de las aplicaciones hechas por el estado. De esta manera se puede tener un mejor control sobre lo que hace el software y evitar posibles puertas traseras.

Una puerta trasera es una funcionalidad oculta en el software que permite hacer cosas que no deberían estar permitidas; por ejemplo en el caso de Ecuador, un afiliado del seguro social (IESS) podría alterar el número de aportaciones para acceder a créditos hipotecarios de una manera rápida e ilegal.

El código fuente de estas aplicaciones críticas debería ser permanentemente auditados para garantizar que no existan puertas traseras. Esta auditoría debería hacerla el gobierno pero además estar abierta para que la sociedad civil pueda realizar todas las auditorías que sean necesarias. De esta manera no solo tenemos que confiar en el auditor que hace el trabajo, sino que cualquier persona que crea que algo se esta haciendo mal pueda acceder al código fuente y reportar estas puertas traseras.

1.3 Gastos Innecesarios

Los instituciones públicas que utilizan software privativo normalmente deben pagar una licencia anual a las empresas desarrolladoras de software por la utilización del mismo. Este cobro que lo suelen hacer empresas transaccionales de manera periódica y anual es muy similar al pago de impuestos. En este caso son impuestos que cobran empresas transnacionales a los gobiernos por utilizar los computadores.

Las licencias de software son por usuario, computadora e institución. Por ejemplo si un ministerio adquiere una licencia para utilizar un sistema de bases de datos privativos esta licencia no podrá ser utilizada por otro ministerio a pesar de que los ministerios pertenecen a un mismo gobierno y todos son parte del estado. Las empresas desarrolladoras de software privativo suelen desarrollar el software una vez y venderlo miles de veces más.

Los gobiernos en lugar de gastar recursos en pagos de licencias a transaccionales de software deberían utilizar estos recursos para migrar a software libre. Este gasto se convertiría en una inversión ya que se lo utilizaría para capacitación, servicios de soporte con empresas nacionales. De esta forma se puede llegar a tener soberanía tecnológica en el software ya que los países dependen de sus propias fuerzas, además que se estará impulsando la industria local de software y estos recursos ya no saldrán del país. Además el dinero invertido en capacitación y educación se convierte en conocimiento de los ciudadanos.

1.4 Beneficios Para el Medio Ambiente

Hace 5 años la mayoría de usuarios de computadora utilizaban las computadoras para lo mismo que hacen ahora: navegar en Internet, revisar el correo electrónico, utilizar procesador de palabras, hoja de cálculo, ver vídeos, escuchar música, etc... ¿Por qué razón un sistema operativo que se supone es moderno como Windows Vista no puede funcionar de manera correcta en computadores de hace 5

años? ¿Donde terminarán todas las computadoras que son totalmente operativas de los gobiernos cuando ya no exista soporte para Windows XP? ¿Nos estamos volviendo productores de basura tecnológica tan solo porque una empresa nos obliga a utilizar hardware más poderoso?

Con el software libre una computadora de hace 5 años sigue siendo funcional y lo será por más tiempo. Es más gracias a proyectos como tcos⁸ o ltsp⁹ computadoras de más de 10 años pueden ser reutilizadas de maneras eficientes. El software libre es mucho más amigable con el medio ambiente que el software privativo.

1.5 Software Público

El software que es desarrollado por un gobierno no es suficiente con que sea software libre sino que además debe ser software público. El software libre no tiene la obligación de ser publicado en Internet, pero el software del estado es desarrollado o adquirido con plata de los contribuyentes y maneja información de los ciudadanos. Es por esto que todos los sistemas desarrollados por el estado deben ser públicos de una manera similar a como lo hace Brasil con el portal de software público¹⁰.

Al ser el software público se garantiza el beneficio de la transparencia, porque todos los ciudadanos tenemos acceso a conocer como funcionan nuestros sistemas; podemos descargar el código fuente del software que utilizamos y auditarlo.

Se beneficia la empresa local de software porque todo software desarrollado por o para el estado esta disponible de manera igual a todas las empresas que deseen proveer servicios alrededor de los sistemas. De esta manera el gobierno tiene la posibilidad de escoger con que proveedor de software va a trabajar

1.6 Integración entre Estados

Desde pequeño crecí soñando con el sueño de Bolívar y una América Latina unida. El el año 2005 este sueño se hizo realidad cuando participé en la organización del primer FLISOL¹¹ donde 10 países de América Latina y más de 100 ciudades se pusieron de acuerdo para ayudar a los demás instalando software libre. Este evento se ha repetido todos los años y en el 2008 participaron más de 200 ciudades y 18 países.

En el caso de los gobiernos esto no debería ser diferente. Todos los gobiernos tienen sistemas y necesidades similares por lo que el desarrollo de estos sistemas sería mucho más eficiente si se lo hace de manera coordinada y se evita duplicar esfuerzos. Por ejemplo Ecuador tiene un sistema de compras públicas hecho completamente en software libre. Este sistema que debe ser libre y público puede ser utilizado por cualquier país de latinoamérica. De esta formas todos ganamos porque se disminuye el costo de desarrollar el sistema y todos los países de la región podemos tener un sistema que permita transparentar las compras públicas, además que si otro país le hace mejoras al mismo estas podrían ser integradas para el resto de países sin problemas.

En la región, Brasil ya ha dado el primer paso con su portal de Software Publico¹², iniciativas similares tiene la Junta de Andalucía¹³ y Venezuela., esperamos que muy pronto Ecuador vaya hacia ese lado y empiece a liberar el código de aplicaciones desarrolladas por y para el gobierno.

8 <http://www.tcosproject.org/>

9 <http://www.ltsp.org/>

10 <http://www.softwarepublico.gov.br/>

11 <http://www.flisol.net/>

12 <http://www.softwarepublico.gov.br/>

13 <http://www.juntadeandalucia.es/repositorio/>

1.7 Software Libre y las Comunidades

El éxito de todo proyecto de software libre radica en la comunidad que hay detrás del mismo. La gran ventaja de un proyecto de software libre es la cooperación que el mismo tiene por la comunidad. Esto reduce los costos de desarrollo, pruebas y además genera un masa crítica para dar soporte al software en las instituciones que lo utilicen.

Si el software del estado se lo licencia con licencias libres y del tipo copyleft¹⁴ como la GPL¹⁵ o AGPL¹⁶ entonces la comunidad juega un rol aún más importante: garantizar que el software sea libre a pesar de que gobiernos futuros tengan políticas distintas. Si el gobierno no es dueño intelectual de todo el código y la comunidad es dueña de parte del código es muy difícil que gobiernos futuros cambien el software a una licencia privativa. Se tendría que convencer a la comunidad para hacerlo o sacar todo el código fuente desarrollada por la comunidad.

1.8 Licenciamiento del Software

El software desarrollado por o para una institución pública debe tener una licencia libre con copyleft. La razón es muy sencilla al ser copyleft todo software desarrollado a partir del original tiene que tener la misma licencia y por tanto ser libre. Si el software tiene una licencia libre sin copyleft entonces una empresa podría tomar ese software para hacer algunas mejoras y cerrarlo para su beneficio propio. Esto no es deseable desde el punto de vista del gobierno que tiene la obligación de beneficiar a la mayoría. No es correcto que unos pocos se aprovechen del trabajo hecho con plata de los contribuyentes.

En el caso de las aplicaciones de escritorio recomiendo utilizar la licencia GPL que es la licencia para software libre copyleft más utilizada. Esto es una ventaja ya que las personas que colaboran en proyectos de software libre están acostumbrados a esta licencia por lo que las reglas de juego están claras y saben que todo su trabajo seguirá siendo libre sin que nadie lo pueda cerrar.

Para las aplicaciones web la GPL tiene una gran debilidad. Si la aplicación no corre en mi máquina sino en un servidor el usuario no tiene derecho a solicitar el código fuente. Para suplir esta falla se creo la licencia AGPL que obliga al desarrollador a entregar el código fuente de una aplicación que corre en un servidor remoto. De esta manera cualquier ciudadano podrá solicitar el código fuente de los sistemas que proveen servicios por parte del gobierno.

1.9 Estándares Abiertos

La forma en la que se graba la información de los ciudadanos es muy importante. Es fundamental que sin importar que aplicación utilicemos en el futuro podamos abrir los documentos actuales. En el mundo del software privativo tradicionalmente los formatos de los documentos van ligados a la aplicación. Normalmente una nueva versión de una aplicación dejará de soportar los formatos antiguos y de esta manera toda la información anterior tendrá que ser migrada al nuevo formato.

Recientemente con el service pack 3 de Microsoft Office 2007 este aplicativo de oficina empezó a bloquear los formatos antiguos de Microsoft¹⁷. Este tipo de comportamiento beneficia solamente a Microsoft ya que los usuarios tendrán que migrar sus documentos al nuevo tipo de formato para que pueda ser leído en el futuro. Si alguien no lo hizo y después de algunos años desea leer sus documentos probablemente no lo pueda hacer. El estado debe garantizar que nuestra información estará disponible en el futuro.

Los estándares abiertos no dependen de una aplicación. Una página web que siga las recomendaciones de la W3C y respete estos estándares se podrá ver bien en todos los navegadores

14 <http://es.wikipedia.org/wiki/Copyleft>

15 <http://www.gnu.org/copyleft/gpl.html>

16 <http://www.fsf.org/licenses/licenses/agpl-3.0.html>

17 <http://support.microsoft.com/kb/938810/en-us>

web que soporten dicho estándar. Una hoja de cálculo hecha con el estándar ODF se podrá ver bien en todas las aplicaciones que soporten este estándar. A diferencia de los formatos cerrados, los formatos abiertos están disponibles para todos por lo que de aquí a 20 años todavía sabremos como se genera un documento hecho con un estándar abierto y de no existir la aplicación para leerlo se la podría desarrollar.

2 Educación y Capacitación

El principal problema que han tenido todos los gobiernos que han querido migrar a software libre es que no existen suficientes personas capacitadas para dar el soporte necesario. Esta situación se la debe atacar en 3 frentes: escuelas y colegios, universidades y profesionales.

2.1 Escuelas y Colegios

En el largo plazo se debe enseñar en las escuelas y colegios computación con software libre. No solo porque los gobiernos utilizaran software libre sino porque con el software libre los niños aprenden a utilizar la computadora y a compartir a la vez.

Por ejemplo si un niño utiliza un programa educativo de software privativo y lo quiere compartir con sus amigo podría estar violando la ley porque la licencia de este programa le dice al niño que compartir es algo malo. Por otro lado si ese mismo programa educativo es libre el niño lo puede compartir con su amigo. En las escuelas se debería enseñar a los niños que compartir es algo bueno y con el software libre se puede hacer esto.¹⁸

La principal ventaja de empezar a enseñar software libre en las escuelas es que por lo general utilizarán el mismo software porque sin importar en que escuela estudie un niño estará aprendiendo más o menos lo mismo según las exigencias del Ministerio de Educación. Esto es muy distinto a lo que sucede en las universidades donde existen innumerables especializaciones donde cada una utiliza otro tipo de software.

Proyectos como Gcompris¹⁹, Tuxmath²⁰, Tuxpaint²¹, entre otros son ejemplos de software libre educativo listo para ser utilizado. No existe software libre que hable sobre la realidad del Ecuador por lo que el Estado debería trabajar para el desarrollo de estos aplicativos. Este software se lo desarrolla una vez y se lo puede utilizar para absolutamente todas las escuelas del país.

2.2 Universidades

Las universidades actualmente enseñan muy poco con Software Libre. Es importante que los estudiantes en las universidades utilicen software libre en todas las carreras y no solo en informática. Por ejemplo un estudiante de diseño gráfico debería aprender con herramientas libres como: Gimp²², Inkscape²³ o Blender²⁴. Al utilizar software libre y no privativo no se obliga a obtener copias ilegales a los alumnos como se hace ahora. Volviendo al ejemplo del estudiante de diseño gráfico tendrá que gastar algunos miles de dólares para tener las herramientas privativas equivalentes a las antes mencionadas. Esto deja a un estudiante con 2 alternativas gastar dinero en las licencias legales o adquirir copias ilegales del software.

En el caso particular de los estudiantes de informática es todavía más importante porque son ellos los que se pueden beneficiar del código fuente. De la misma manera que un estudiante de literatura

18 Recomiendo leer este artículo de Richard Stallman sobre el software libre y la escuela:

<http://www.gnu.org/philosophy/schools.es.html>

19 <http://gcompris.net/-es->

20 <http://sourceforge.net/projects/tuxmath/>

21 <http://www.tuxpaint.org/>

22 <http://www.gimp.org/>

23 <http://www.inkscape.org/?lang=es>

24 <http://www.blender.org/>

aprende a escribir leyendo libros de los mejores escritores o un estudiante de música lo hace leyendo partituras de los mejores músicos.²⁵ Los estudiantes de informática deben aprender a programar leyendo código fuente de grandes desarrolladores de manera similar a como lo hacen los estudiantes de literatura. Esto se puede lograr solamente con el software libre ya que si un estudiante quiere saber como esta hecho un procesador de palabras puede descargar el código fuente del OpenOffice.org Writer²⁶ o de Abirword²⁷, algo que no se puede hacer con Microsoft Word.

Para lograr que los estudiantes salgan de las escuelas manejando software libre es muy importante que los profesores dominen el tema. Es por esto que una capacitación a los profesores universitarios es fundamental para poder obtener el éxito en la implementación de software libre en las universidades. El Master de Software Libre de la Universidad Abierta de Cataluña²⁸ es un paso importante que pueden dar los profesores para aprender a utilizar y entender el software libre.

2.3 Capacitación a Profesionales

Tanto los funcionarios públicos como las empresas privadas necesitan capacitarse en software libre. Los funcionarios para poder administrar los sistemas existentes y las empresas para poder proveer servicios sobre estos.

En el caso particular de América Latina donde Ecuador, Brasil, Venezuela y Paraguay tienen políticas públicas que favorecen la utilización de software libre, se debería crear un instituto en línea donde los mejores profesionales puedan dar capacitación a gente de toda América Latina. Esta es una forma muy económica y rápida en la que los profesionales pueden empezar a capacitarse.

La capacitación presencial se la debería hacer de forma piramidal donde la prioridad la tengan los profesores. La idea es capacitar a capacitadores para que estos repliquen lo aprendido a más gente.

3 Conclusiones

El software libre es la única opción que tienen los estados para ser soberanos en cuanto a software se refiere. Si queremos depender de nuestras fuerzas debemos tener el conocimiento de como funciona lo que estamos utilizando y en el caso del software el código fuente nos da esta oportunidad. Todo el software desarrollado por y para el gobierno a más de ser libre debe ser público para garantizar su transparencia y brindar al sector privado la oportunidad de generar servicios sobre el mismo.

Para tener el éxito y poder migrar a software libre de una manera exitosa la capacitación es fundamental. Esta capacitación se la debe hacer en todos los ámbitos de la sociedad, desde los niños en las escuelas, los jóvenes en la universidad y los profesionales que deben actualizarse.

25 Esta idea la escuché a Federico Mena en el video semillas de Libertad: <http://www.kazak.ws/semilla/semilla.php>

26 http://es.wikipedia.org/wiki/OpenOffice.org_Writer

27 <http://www.abisource.com/>

28 http://www.uoc.edu/masters/oficiales/master_oficial_software_libre/master_oficial_software_libre_plan.htm