

Condiciones Técnicas, Legales e Institucionales para promover las Tecnología Libres en Venezuela

Condiciones Técnicas, Legales e Institucionales para promover las Tecnología Libres en Venezuela	1
Introducción	1
Desarrollo	2
Cronología o momentos del Software Libre	2
Aspecto formal y no formal del Software Libre y Software Abierto	3
Valores que se expresan en el Software Libre (SL)	6
Ventajas del Software Libre (SL) en la Administración Pública (AP)	7
Ventajas del SL en la administración pública	7
Desventajas del SL en la administración pública	8
Buscando comprender la ideología del Software Libre: Antimonopólico Capitalista - Reformista - Socialista.	8
Propuesta de Continuación del Proyecto.....	11
Referencias.....	11

Por:

Alberto Medrano, amedrano@cenditel.gob.ve

Ángel Serra, aserra@cenditel.gob.ve

Giancarlos Colasante, gcolasante@cenditel.gob.ve

Pedro Buitrago, pbuitrago@cenditel.gob.ve

Introducción

En ánimos de buscar modos para lograr la soberanía y la apropiación tecnológica del Hardware y Software que involucre a las comunidades activamente a la atención de los problemas que se manifiestan en diversidad de formas en la sociedad, y donde hay cavida a posibles soluciones que puedan hacerse desde las tecnologías libres, se espera con esta investigación lograr unas políticas consistentes que permitan la apropiación del conocimiento con un sentido más crítico e enriquecedor de estas tecnologías libres, considerando que las tecnologías libres y los movimientos que las sustentan se logra en la práctica cuando ponen en funcionamiento los valores que las tecnologías libres expresan, en base a esto nos planteamos la siguiente pregunta como tema de investigación para el taller de la especialización en Sistemología Interpretativa, ¿cuales son las condiciones técnicas, legales e institucionales para promover el software y hardware libre en Venezuela?.

Cabe destacar que un cuestionamiento de tal magnitud debe considerar la dinamica de la nación, tambien es necesario estudiar como estos factores se consiven y orden cronologico que nos permitan ver un panorama de como se consiven las tecnologías libres y como son en la actualidad, el sentido de la pregunta redica que no puede simplemente importar un modo de hacer, sino adaptarlo y modificarlo a modo de que sea útil para la sociedad.

El identificar el porque las tecnología libres tambien se asocian con otros movimientos sociales y políticos que no tienen que ver directamente con el software, el planteamiento de la pregunta debe dar luces hacia lo que podria ser una propuesta que pueda ser llebada a una instancia para su evaluación e implementación como futura política de estado que

pueda ser camino a lograr cierta independencia tecnológica que nos vuelva más independientes y soberanos.

Desarrollo

Cronología o momentos del Software Libre

Un intento para ver y entender en mejor manera como se consolidan las tecnologías libres, software libre (free software) y software abierto (open-source), lo presentamos a modo de momentos, tenemos entonces:

1. Concepción
2. Privatización
3. Revelación
4. Oportunidades

Resumido en la siguiente tabla:

Momento /Acción	Reseña	Fecha de Aparición y Actores Principales
1 Concepción	Este momento comprende la concepcion de la tecnologia de software y los primeros desarrollos de lenguajes de programación, se acepta una nueva era de desarrollo científico y tecnologico al incorporar la computación y los circuitos logicos avanzados. Los desarrollos eran libres y se compartian entre pares en Universidades destacadas, la comunidad era muy selecta, los equipos eras demasiado costozos y no eran portables	<ul style="list-style-type: none"> • (1936). Konrad Zuse, diseño y fabricó la Z1, Primer Computador Programable (electro-mecánico). • (1947). ENIAC. Primera computadora digital electrónica. • (1958). Jack Kilby, Desarrolla el primer Circuito Integrado.
2 Privatización	Al mejorar la escala de integración de los circuitos electrónicos, mejorar las capacidades de los ordenadores, disminución de su coste para su venta masiva al publico, se ven oportunidades de negocio en la venta de software para uso masivo en los nuevos ordenadores de tipo personal o PC (Personal Computer), se privatiza el conocimientos al dar los programa no en su forma fuente sino en su forma binaria listo para se ejecutado por el ordenador.	<ul style="list-style-type: none"> • (1975). Bill Gates y Paul Allen, constituyen Microsoft Corporation • (1976). Steve Jobs y Steve Wozniak, constituyen Apple Computer, Inc • (1981). IBM presenta la plataforma IBM PC Compatible. Fue llamado IBM 5150. • (1985). Lanzamiento de la primera versión del Sistema Operativo Microsoft Windows
3 Revelación	Un grupo de individuos en contra de la privatización del conocimiento en software y las corporaciones que sustenta esa modalidad, manifiestan y	<ul style="list-style-type: none"> • (1983). Richard Stallman, aparece con el proyecto GNU y el

	<p>consolidan una plataforma para el desarrollo de software libre, luego se suman otras iniciativas como software abierto que de acceso fuentes de los programa y proponer mejoras de estos o simplemente usarlos.</p> <p>El software libre consolida nuevas formas de desarrollo soportados por una comunidad distribuida a nivel mundial.</p>	<p>proyecto de Software Libre</p> <ul style="list-style-type: none"> • (1991). Linus Torvalds, desarrolla el primer kernel (núcleo) basado en UNIX • (1992). Se logra un sistema operativo completo dando lugar a las distribuciones GNU/Linux • Stallman apoya al proyecto de la Free Software Foundation (FSF), Torvalds apoya al proyecto de la Open Source Initiative (OSI)
<p>4 Oportunidades</p>	<p>Nuevas relaciones se crean en esta dinamica de desarrollo de software libre y abierto, algunas corporaciones en diversas formas ven oportunidades de negocio y participan, éste trabajo conjunto ha logrado proporcionalmente dar a conocer la comunidad de software libre y ha habido beneficio mutuo.</p>	<ul style="list-style-type: none"> • (1999). Sun Microsystems adquiere a StarDivision (compañía alemana) por 73,5 millones de dolares y con ella la aplicación ofimática StarOffice, cuya versión gratuita fue entrega en junio de 2000 y relanzada como OpenOffice.org en octubre del mismo año.

Aspecto formal y no formal del Software Libre y Software Abierto

Las tecnologías libres y todos los involucrados en ella puede manifestarse en dos modos:

- Modo Formal
- Modo No Formal

Y en ella hay cabida tanto para usuarios, desarrolladores, organizaciones, fundaciones, corporaciones, empresas, gobierno, etc.

A continuación presentamos las directrices a cumplir para que los desarrollos sean amparados bajo las figuras de los siguiente movimientos:

En un estado "formal" tenemos aquellas que son definidas por la Free Software Foundation (FSF) y las cuatro libertades del Software Libre.

1. La libertad de **usar** el programa, con cualquier propósito.
2. La libertad de **estudiar** cómo funciona el programa y modificarlo, adaptándolo a tus necesidades.
3. La libertad de **distribuir** copias del programa, con lo cual puedes ayudar a tu prójimo.
4. La libertad de **mejorar** el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.

Otro aspecto "formal" es el definido por la Open Source Initiative (OSI) y las 10 premisas

del Software Abierto.

1. Libre redistribución: el software debe poder ser regalado o vendido libremente.
2. Código fuente: el código fuente debe estar incluido u obtenerse libremente.
3. Trabajos derivados: la redistribución de modificaciones debe estar permitida.
4. Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.
5. Sin discriminación de personas o grupos: nadie puede dejarse fuera.
6. Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.
7. Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa.
8. La licencia no debe ser específica de un producto: el programa no puede licenciarse solo como parte de una distribución mayor.
9. La licencia no debe restringir otro software: la licencia no puede obligar a que algún otro software que sea distribuido con el software abierto deba también ser de código abierto.
10. La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

Y parecida a la anterior pero con sus diferencias, podemos nombrar a las directrices de software libre de Debian (DFSG).

1. Libre redistribución: La licencia de un componente de Debian no puede restringir a un tercero el vender o entregar el programa como parte de una distribución mayor que contiene programas de diferentes fuentes. La licencia no debe solicitar «royalties» u otras comisiones para su venta.
2. Código fuente: El programa debe incluir el código fuente completo, y debe permitir la distribución en forma de código fuente y en forma compilada (binario).
3. Trabajos derivados: La licencia debe permitir modificaciones y trabajos derivados y debe permitir que estos se distribuyan bajo los mismos términos que la licencia del programa original.
4. Integridad del código fuente del autor: La licencia puede restringir la distribución del código fuente en forma modificada "sólo" si la licencia permite la distribución de "parches" ("patch files") para poder modificar el código fuente original del programa en el momento de compilarlo.
5. No discriminación contra personas o grupos: La licencia no debe discriminar a ninguna persona o grupo de personas.
6. No discriminación en función de la finalidad perseguida: La licencia no puede restringir el uso del programa para una finalidad determinada. Por ejemplo, no puede restringir el uso del programa a empresas con fines comerciales, o en investigación genética.
7. Distribución de la licencia: Los derechos y libertades de uso asociados al programa deben aplicarse en la misma forma a todos aquellos a los que se redistribuya el programa, sin necesidad de pedir una licencia adicional para estas terceras partes.
8. La licencia no ha de ser específica para Debian: Los derechos asociados al programa no deben depender de que el programa sea parte o no del sistema Debian. Si el programa es extraído de Debian y usado o distribuido sin Debian, pero manteniendo el resto de las condiciones de la licencia, todos aquellos a los que el programa se redistribuya deben tener los mismos derechos que los dados cuando forma parte de Debian.
9. La licencia no debe contaminar a otros programas: La licencia no debe poner restricciones sobre otros programas que se distribuyan junto con el programa

licenciado. Por ejemplo, la licencia no puede insistir que todos los demás programas distribuidos sobre el mismo medio deben ser software libre.

10. Ejemplos de licencias: Las licencias "GPL", "BSD", y "Artística" son ejemplos de licencias que nosotros consideramos "libres".

Revisando la definición del software libre de la FSF y los manifiesto del OSI y DFSG llegamos a la conclusión que las directrices más pertinentes para lo que deseamos construir para el socialismo venezolano son las de la FSF.

La siguiente tabla resume este aspecto formal y no formal del software libre, teniendo en cuenta que la parte formal corresponde a la definida por la FSF y la no formal es la que se da en la praxis desde las comunidades de software libre.

Aspecto Formal	Aspecto No Formal
<p>Aquel que es definido por la Free Software Foundation (FSF)</p> <ul style="list-style-type: none"> • 4 Libertades: • la libertad de usar el programa, con cualquier propósito. • la libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a tus necesidades. • la libertad de distribuir copias del programa, con lo cual puedes ayudar a tu prójimo. • la libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie. <ul style="list-style-type: none"> • No se pregunta más allá de esas 4 libertades, no existe una actitud crítica hacia lo que la FSF dice. • FSF = Capitalista en defensa de la individualidad. • Apunta a la neutralidad tecnológica 	<ul style="list-style-type: none"> • Desde las Comunidades • Se preguntan por su rol en la sociedad • Se trabaja de forma aislada o forma parte de un proyecto país. • Se pregunta sobre la ética de la tecnología. • Se pregunta sobre la conciencia <p style="text-align: center;">CONCIENCIA + ÉTICA</p>

En el sentido del tecnologías libres, no se discuten aspectos como ética y conciencia sobre el desarrollo de tecnologías, se presenta como la libertades en un sentido meramente individual, es decir, la necesidad de los individuos hacia el software libre. El software libre es el ser en contra del software privativo y no de las corporaciones y empresas en busca de capital, sin embargo muy marcado en contra del monopolio en el área de las tecnologías de software.

Estas formas de organización (empresas, corporaciones, etc.), hacen vida en el software libre, casos por ejemplo (proyecto + patrocinante):

- OpenOffice, Sun Microsystems ahora comprado por ORACLE.
- Ubuntu, Canonical Corporation.
- openSUSE & SUSE, Novell Corporation.
- Fedora & RedHat, RedHat Corporation.
- Firefox, Mozilla Corporation.

- CANAIMA GNU/Linux, Gobierno de la Republica Bolivariana de Venezuela.

Valores que se expresan en el Software Libre (SL)

El movimiento del SL no sólo defiende las libertades individuales (expresadas en las 4 libertades) sino que busca crear conciencia bajo la argumentación de la ética y la moral que implica tener el conocimiento, y en consecuencia la posibilidad de ayudar a los demás. Para discutir debemos preguntar ¿que entendemos por ética? Sin hacer la distinción formal que en filosofía se hace entre ética y moral, podemos denominarla como el conjunto de creencia que opera como guía del obrar, con la que se distingue el bien y el mal obrar [1], y entre esas creencia están los valores.

La Figura 1 muestran los valores que se expresan en el SL de acuerdo al mapa conceptual de GNU [2].



Figura 1. Valores expresados en el software libre. Fuente: wikipedia.

En esta discusión se puede caer en silogismos erróneos con facilidad, como afirmar el SL es socialisata, o que el SL es capitalista. Las 4 libertades, que defienden derechos individuales, por sí solas no propician ni fortalece la conciencia ni la ética que busca el MSL (movimiento del Software Libre) en su ámbito no formal. Más bien las 4 libertades apoyan el individualismo, para entender esta idea ilustremos un ejemplo: al compartir un programa todos los que lo usen y estudien están en la misma condición, pero aquella persona que lo modifique, agregándole una funcionalidad estará en ventaja respecto al resto, las 4 libertades permiten pero no obligan compartir. Siendo el individuales uno de los valores propios del capitalismo, se puede pensar que las 4 libertades apoyaría el capitalismo, su carácter anti-monopólico lo demuestra.

Ahora encontramos un error que ocurrir con cierta frecuencia, afirmar que el SL es socialista. ¿Qué sustenta esta afirmación?, sin ahondar en aguas profundas vemos que la solidaridad es una de las piedras que apoya tal afirmación. Ciertamente es un valor del socialismo, sin embargo, ¿se entiende la solidaridad del mismo modo en el SL y en el socialismo? Parecen estar en dimensiones distintas, entiéndase que la solidaridad en el SL está delimita a compartir código, en cambio en el socialismo tiene mayores implicaciones, tocando aspectos que van más allá del ámbito del trabajo. Una perspectiva similar acontece con la eficiencia, ya que en el capitalismo se tiene una concepción -cantidades y ganancia- distinta a la que se tiene en el socialismo -calidad, un buen trabajo para que los demás se beneficien-.

Al SL no se le pueden atribuir los valores, ya que el software no puede tener valores, son las personas las que tienen valores. En ese caso podemos decir que el SL propicia ciertos valores en el conjunto de personas que la utiliza, y a su vez dentro de las comunidades de

SL cada comunidad cultivará un conjunto de valores, muchos compartidos entre las diferentes comunidades pero no en todos los casos. Por ello se ven concepciones tan distintas sobre el software libre como la FSF y la OpenSource, o el caso Debian y Ubuntu.

Ventajas del Software Libre (SL) en la Administración Pública (AP)

En nuestra investigación nos concentramos en las ventajas y desventajas de la aplicación del software libre en la administración pública debido al fuerte apoyo que existe por parte del gobierno en tal movimiento, ejemplo de ello es el Decreto 3390 de 23 de diciembre de 2004.

Algunos aspectos extraídos de: [http://www.cnti.gob.ve/index.php?option=com_content&task=view&id=94]

Ventajas del SL en la administración pública

- Bajo costo de adquisición, administración e implementación:
 - Las Instituciones de la AP que adquiere aplicaciones de SL lo hace sin ninguna erogación monetaria o a muy bajo precio, liberándose del pago de licencias que en su mayoría son costosas y les limita el uso de la aplicación
 - La adquisición de aplicaciones de SL concede la libertad de instalar las aplicaciones el número de veces y máquinas que requiera, sin tener que realizar pagos adicionales.
 - La mayoría de soluciones en SL por su concepción modular (por ejemplo los Sistemas Operativos) no requieren de mucho poder de procesamiento de las máquinas para utilizarlo, solo basta en instalar los módulos que requieres.
 - En las diferentes instituciones de la AP, existen aplicaciones de igual objetivo y propósito que permite la reusabilidad de las aplicaciones.
- Independencia del proveedor:
 - Si las instituciones AP tiene personal con conocimientos adecuados puede corregir errores, modificar la aplicación, desarrollar nuevos módulos de acuerdo a sus necesidades, librándose de depender del proveedor de la aplicación, en caso de no contar con personal calificado, puede acudir a terceros (personas, instituciones, empresas) con conocimiento adecuado que le ofrezca desarrollo o servicio a la aplicación. Aparte permite fomentar el desarrollo local.
 - Libre acceso a la información pública, ya que maneja formatos y protocolos estándar, permitiendo la interoperabilidad entre las aplicaciones de las diferentes instituciones de la AP
- Seguridad Nacional:
 - Fortaleciendo la seguridad nacional, ya que las instituciones públicas con el SL tienen el control por completo de los procesos de flujo, almacenamiento y modificación de información personal de los venezolanos
 - Las instituciones AP pueden verificar el procesamiento de datos para saber si la aplicación cumple su funcionamiento de acuerdo a lo que promete hacer.
 - Chequear si las vulnerabilidades existentes en la aplicación son intencionales o accidentales
 - Al exponer constantemente las aplicaciones a procesos de auditoría, hace más difícil ocultar funciones anormales como por ejemplo: puertas traseras entre otros, haciéndola más segura.

Desventajas del SL en la administración pública

- Las aplicaciones no son muy conocidas
 - En el momento de que usarla, pueden generar rechazo automático cuando de presenta una solución alternativa a la que tiene acostumbrado a utilizar
 - No son consideradas en el momentos de implementación de soluciones
- Faltas de decisiones políticas
 - Falta de acompañamiento a las políticas publicas en la masificación del uso del SL,
- Técnicas
 - las mayoría de la configuraciones de hardware y software no es intuitivo y aparte la falta de una buena documentación, se requiere de ciertos conocimientos previos,
- Culturalmente
 - Estamos acostumbrados por nuestro sistema económico rentista, a soluciones lista para ser usadas y el SL puede requerir algún tipo de esfuerzo para al menos su utilización.
 - Debido a que grandes transnacionales invierten dotando laboratorios con sus productos a las universidades, esta termina por formar a estudiante administradores de esas productos privativos

Buscando comprender la ideología del Software Libre: Antimonopólico Capitalista - Reformista - Socialista.

EL movimiento de software libre a sido tildado de capitalista hasta de comunista, existen desarrolladores pertenecientes a la comunidad con posturas bastante diferentes en el ámbito político, "...En las comunidades de desarrolladores y usuarios de software libre es frecuente encontrar individuos de derecha ya que consideran que el software libre es la máxima expresión del capitalismo como sistema de libre comercio, sinónimo a libre mercado, mas allá de las barreras de gobiernos y monopolios." "...Christian Imhorst en su artículo "Anarchy and Source Code – What does the Free Software Movement have to do with Anarchism?"²⁹" establece la analogía entre el anarquismo y el desarrollo del software libre, alegando que este es escrito de forma voluntaria y no es regulado de manera jerárquica." [3] por lo que el tema de si el software libre apoya un movimiento político en particular ha sido algo confuso, por ello tendríamos que hacernos la siguiente pregunta. ¿El de Software Libre Posee Ideología?

Eduardo Samán nos expone:

"El Software libre no es un fin en si mismo, es un medio, una herramienta tecnológica, por tanto la ideología asociada al mismo viene dada por el modo de su producción o su uso. " [3]

"...como instrumento tecnológico el software libre ¿qué ideología tiene? ¿Es capitalista o socialista? Ningún instrumento en sí tiene ideología; ella está asociada a la forma como se produce." "el software libre no tiene una ideología determinada, pero depende de cómo se lo produce y para qué se lo usa." [4]

Lo anterior tiene mucho sentido y aclara bastante la situación, por lo que podemos decir que la ideología del software se la dan las personas que lo producen y lo utilizan, donde podemos conseguir a la Free Software Foundation como líder de los movimientos de software libre, pasando por la iniciativa Open Source, hasta llegar a gobiernos, anarquistas y comunistas.

"El movimiento del software libre que lidera Stallman, inicia el desarrollo del software en

base sólo a tres principios: antimonopólico, solidario y libertario." [3]

El hecho de que el software libre sea antimonopólico no implica que sea capitalista, ya que el monopolio atenta a la libertad de mercado, la libre competencia y la diversidad, los cuales conocemos como características del capitalismo. Entendemos como monopolio cuando una empresa o industria posee el control de un producto en el mercado o en otras palabras es la situación de mercado en que la oferta de un producto se reduce a un solo vendedor. En el mundo del software, la principal referencia es Microsoft Corporation, la cual fue objeto de acciones antimonopolíticas por la parte del gobierno de Estados Unidos, considerado como principal exponente del capitalismo, a pesar de que este conduce inevitablemente a una acumulación de capital, condena a los monopolios lo cual pudiese verse algo contradictorio.

El software libre siempre ha sido etiquetado como principal enemigo de la corporación Microsoft o en oposición a este a pesar de que muchas comunidades de software le molesta tal denominación, pues manifiestan que el software libre va más allá de ser una simple acción antimonopolio. Por otro lado tenemos la iniciativa del Open Source la cual es tildada de ser más capitalista que la del software libre. *"El movimiento de "Software Libre"; de Richard Stallman esta "más a la izquierda"; que el "Open Source"; de Linus Torvalds, el cual descaradamente pondera el mercado capitalista." [5].* En la iniciativa Open Source conseguimos muchas empresas o corporaciones capitalista como IBM, Novell, Sun Microsystems porque lo vieron la oportunidad de negocio, bajos costos de producción y la forma de competir con Microsoft.

El software libre también se le ha visto como reformista basándose en los tres principios mencionados anteriormente, Eduardo Samán refuta muy bien este hecho.

"Hay otra corriente en esto, que es la reformista. Ahí consideran sólo tres principios: por un lado, es antimonopólica. Eso, en sí mismo, no la hace anticapitalista ni revolucionaria. El capitalismo, dentro de su hipocresía, de su doble moral, llega a condenar el monopolio. Ven a Microsoft como el malo de la película porque interfiere en el libre juego de la oferta y la demanda. Pero el capitalismo puede ser compatible con el software libre. Esta corriente tiene como otro principio el hecho de ser de producción solidaria. Pero esto sólo, por sí mismo, no es anticapitalista. Por ejemplo las cooperativas son estructuras de producción solidaria, pero eso no significa socialismo. Son más democráticas en la distribución de las riquezas que una sociedad anónima, eso sin dudas; pero aún no son socialistas en sentido estricto. El tercer principio es el ser libertario. Eso hace alusión a la libertad que confiere para copiarlo, distribuirlo; es decir, la libertad de disposición del software. Pero eso es la libertad de la estatua de la Libertad y no es la libertad que nosotros entendemos que debe construir la revolución." [4], en ese mismo entrevista Samán alega que el software libre no puede ser reformista si sigue las leyes del mercado y no vas allá en la búsqueda de la transformación de una sociedad. *"el software libre tiene que dar ese paso para transformarse en un instrumento que sirva a la liberación, saltar sus versiones corporativas o socialdemócrata, que no pasa de un reformismo que siempre sigue en el ámbito del mercado." [4]*

Reforzando la idea de que el software libre no puede ser reformista porque esta involucrado en el juego del mercado tenemos a Leonardo Vazquez en el artículo *"El software libre y sus límites bajo el capitalismo"* publicado en www.rebellion.org.

"...Desde la versión reformista de Stallman y sus amigos, que busca mostrar a la lucha por el software libre como una lucha por humanizar al capitalismo mediante reformas para lograr un mundo "mejor", en donde Microsoft es el malvado que obstruye la sana lógica del libre mercado." [6]

Finalmente tenemos al software libre como socialista, donde conseguimos como principal premisa para este tiene que ser antiexplotador, como se dijo anteriormente la ideología del software libre va depender de la forma de su uso y producción.

Eduardo Samán lo expone bastante claro:

"Si el software libre, como herramienta, se produce y usa bajo una modalidad capitalista entonces será capitalista. Para que el software libre se convierta en socialista no debe mediar una relación de explotación durante su desarrollo o uso." [3]

Reforzando esta idea podemos citar algunos fragmentos del artículo de Sandra Parra publicado en *aporrea*:

"Que pueda haber relaciones de producción del tipo capitalista en el 1% de las empresas que desarrollan software libre, no es algo propio del software libre, sino del sistema económico-social imperante. En las sociedades capitalistas, generalmente las relaciones de producción de "cualquier bien" que se produzca, son de explotación, en las que pocos dueños se apropian de la plusvalía de sus trabajadores."

"Los más destacados del software libre, es que los conocimientos que lo conforman son libremente accesibles por todos los individuos de una sociedad, y es precisamente esta característica, por lo que se destaca como el bien más socialista que haya producido alguna vez el mundo."

"No hay límites para las ganancias derivadas de la explotación de un producto final de software libre, éste es un argumento muy usado por los que dicen que el software libre es capitalista, sin embargo, al ser un bien público, de libre acceso, los especuladores no pueden abusar: "en una sociedad consciente de estos bienes", porque serían reprendidos moralmente por esta. El problema está, entonces, en que la sociedad se eduque y conozca los bienes "libres" que produce, para que su contraloría social haga el trabajo con efectividad" [7]

Finalmente y en búsqueda de esas condiciones para fomentar el desarrollo de un software libre acorde al socialismo venezolano citamos los siguientes fragmentos:

"En el proceso bolivariano entendemos que la verdadera libertad es cuando se alcanza la igualdad entre los seres humanos y esto pasa por la desaparición de las relaciones de explotación, consideramos que el software libre, independientemente de su ventaja tecnológica intrínseca, es una buena herramienta para romper con el modelo de explotación capitalista en la etapa transitoria para la construcción del Socialismo del Siglo XXI. Esto es porque el software libre: no es propiedad privada sino colectiva, no es individualista sino comunitario, no se basa en la competencia sino en la cooperación, no requiere de capital ni compañías para existir, rompe el paradigma de la propiedad intelectual como requisito sinequanon para la innovación.

En la etapa de transición, es necesario rescatar el software libre de la elite intelectual y popularizarlo para hacerlo también antiexplotador. Debemos elegir usar software desarrollado con iniciativas no comerciales o contratar la elaboración de software a cooperativas o empresas de producción social (EPS)³¹. Así también, debilitar el sistema de propiedad privada sobre el conocimiento.

En conclusión el Software Socialista debe ser: libre de capital, libre de compañías, libre del mercado, libre de la propiedad, libre de burocratismo y libre de explotación." [3]

"Los socialistas nos apropiamos del Software Libre para la liberación y soberanía tecnológicas de nuestros pueblos, pero no debemos caer en la trampa de seguir ideologías preelaboradas como el "softwarelibrismo" o stallmanismo que reducen la lucha a lo netamente técnico "código libre o no" mientras todos los militantes por un mundo mejor y más justo, conocemos bien las calles y tal reduccionismo sería funcional o hacer el caldo gordo al imperialismo. Los socialistas debemos seguir por el Software Comunitario de carácter clasista y no liberal." [5]

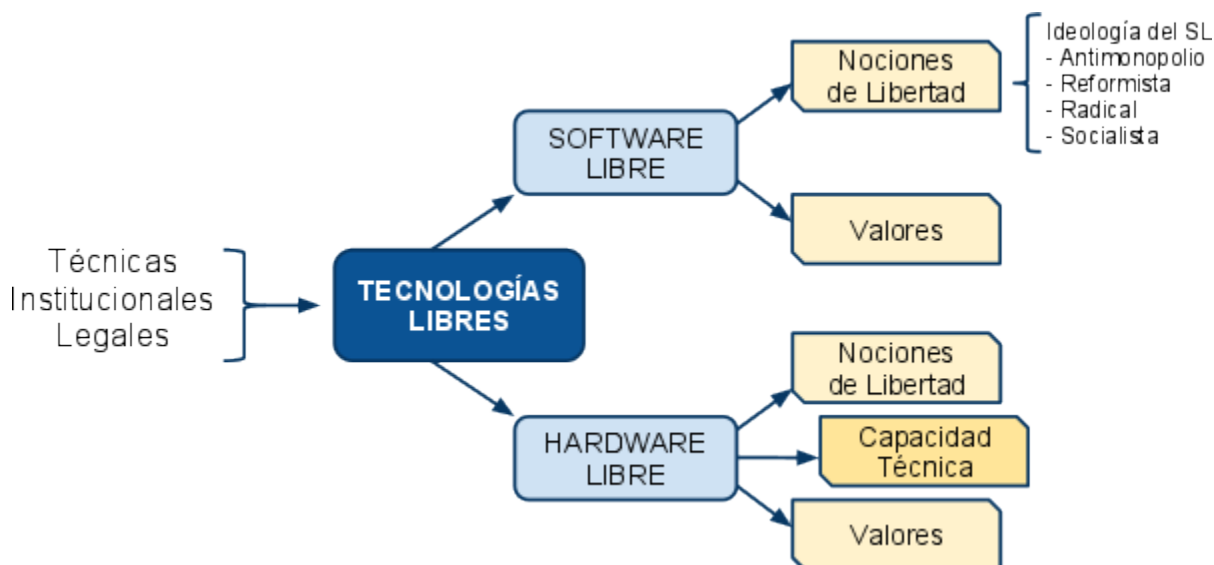
"... Ustedes también son proletarios, nadie se asombre; son proletarios del trabajo intelectual, proletarios del pensamiento, proletarios de las ideas, proletarios en la elaboración de mensajes; son proletarios, incluso, cuando salen corriendo a llevar un

reportaje al periódico, y antes, cuando no existían las computadoras, tecleando desesperadamente en una máquina de escribir. Son, además, obreros asalariados." Fidel Castro [5]

Propuesta de Continuación del Proyecto

Se puede entonces investigar y validar sobre las condiciones Técnicas, Institucionales y Legales para la implementación del Software Libre y más profundamente las tecnologías de Hardware Libre, que sin más, el que hacer, los requerimiento y modalidades de trabajo difieren en cierta manera respecto al Software Libre.

Tenemos como fin plantear cuales son las condiciones técnicas, institucionales y legales para la incorporación de la tecnologías libres en Venezuela en congruencia con el proyecto socialista de país que se desea construir donde se promueva la difusión del conocimiento, la construcción colectiva, un cambio en las relaciones sociales, donde prive el bien del colectivo, la libertad, la solidaridad y el respeto además de promover la formación, desarrollo y organización de colectivos y comunidades. A continuación presentamos el esquema que hemos estado siguiendo en nuestra investigación.



Referencias

- [1] <http://es.wikipedia.org/wiki/Moral>
- [2] http://es.wikipedia.org/wiki/Software_libre
- [3] Eduardo Samán (2007): Del Software Libre al Software Socialista. <http://www.aporrea.org/tecno/a29113.html>
- [4] Entrevista Eduardo Saman: Para que el software libre sea tal, tenemos que romper con las relaciones de explotación. encontrarte.aporrea.org/media/46/hablando46.pdf
- [5] Troyano (2009): Las diferencias de los socialistas revolucionarios con el Software Libre propuesto por Richard Stallman <http://www.rebellion.org/noticia.php?id=80924&titular=las-diferencias-de-los-socialistas-revolucionarios-con-el-software-libre-propuesto-por-richard-stallman->
- [6] Leonardo Vazquez: El software libre y sus límites bajo el capitalismo. <http://www.rebellion.org/noticia.php?id=989>

[7] Sandra Parra (2007): El software libre es socialista. <http://www.aporrea.org/tecnologia/a40270.html>