

III Congreso Latinoamericano De Educación Superior “LATINOAMÉRICA EDUCA”

Bloque Temático 4: “La integración Estratégica para nuestros Pueblos”

Mesa 11: “El Software libre y manejo soberano de la información”

Introducción y Marco

Mi nombre es **Juan Carlos Stecher** y como coordinador de esta mesa quiero hacer una introducción al tema del Software Libre, en beneficio del público que nos acompaña.

Además estableceré la relación entre el debate sobre Software Libre y este congreso. Que interés tiene este debate, a mi entender, para los objetivos de este congreso.

Para los que desconozcan el tema, se denomina software, a todos los componentes intangibles de una computadora, es decir, al conjunto de programas y procedimientos necesarios para que dicha computadora pueda realizar una tarea. Esto incluye aplicaciones como puede ser un procesador de palabras, o una hoja de cálculo, y software de sistema como un sistema operativo, que permite al resto de programas funcionar adecuadamente, realizando la interacción con los componentes físicos y el resto de aplicaciones.

Otro tema importante es que, por lo general, el programa que es escrito por seres humanos, y que estos pueden leer y entender, no es el mismo código que se ejecuta en la computadora.

Para poder ejecutar un programa es necesario, por lo tanto, convertir ese programa que escribió un programador (denominado CODIGO FUENTE del programa), a algo que pueda ser ejecutado por el “*cerebro*” de la computadora. Esta tarea es realizada por algo llamado compilador y lo que se obtiene se denomina CODIGO OBJETO del programa.

Este tema del CODIGO FUENTE y su correspondiente CODIGO OBJETO tiene importancia, en la libertad que puede establecerse para poder introducir mejoras en los programas que componen un determinado Software. Si bien existen procesos de ingeniería reversa que permitirían a veces obtener el CODIGO FUENTE desde el OBJETO, suele ser una tarea extremadamente laboriosa, inexacta y no fácilmente accesible.

Software Libre es un movimiento mundial, originado por el norteamericano Richard Stallman en la década de 1980.

Hay un libro de Stallman titulado “*Software libre para una sociedad libre*” del cual he tomado algunos párrafos para esta introducción. Recomiendo su lectura para entender el tema del *Software Libre*. Está disponible en Internet y es de libre distribución.

También he tomado algunas definiciones de Wikipedia, que es el mayor proyecto de enciclopedia de contenido libre, y esta ubicada en Internet.

Cuenta Stallman que en la década de 1970 trabajaba en el Laboratorio de Inteligencia Artificial del MIT (Instituto Tecnológico de Massachussets), como desarrollador de sistemas operativos:

"...Cuando entré a trabajar en el Laboratorio de Inteligencia Artificial (AI Lab) del MIT en 1971, pasé a formar parte de una comunidad que compartía software y llevaba haciéndolo durante años. El acto de compartir software no se circunscribe a nuestra comunidad en particular: es tan antiguo como los propios ordenadores, lo mismo que compartir recetas es tan viejo como la cocina. Simplemente, nosotros lo hacíamos en mayor medida..."

"...No llamábamos «software libre» a nuestro software porque el término no existía todavía; pero era exactamente eso..."

Estos programas desarrollados y compartidos en la comunidad se ejecutaban en una computadora denominada PDP-10 bajo el nombre ITS (Incompatible Timesharing System). Pero a principios de 1980 esta comunidad se derrumba, cuando la mayor parte de los programadores son contratados por una empresa privada, el PDP-10 deja de fabricarse, y toda la programación del ITS queda obsoleta e inutilizable en las computadoras nuevas. Dice Stallman en diferentes párrafos:

"...De esa manera se enterraba definitivamente al ITS: quince años de trabajo tirados por la borda..."

"...Los modernos ordenadores de la época, como el VAX o el 68020, contaban con su propio sistema operativo, pero ninguno utilizaba software libre. Había que firmar un acuerdo de confidencialidad incluso para obtener una copia ejecutable. Todo ello significaba que antes de poder utilizar un ordenador tenías que prometer no ayudar a tu vecino. Quedaban así prohibidas las comunidades cooperativas. Los titulares de software propietario establecieron la siguiente norma: «Si compartes con tu vecino, te conviertes en un pirata. Si quieres hacer algún cambio, tendrás que rogárnoslo»..."

"...Una vez desapareció mi comunidad, era imposible seguir como hasta entonces. De modo que me enfrenté a un dilema moral radical. Lo más fácil hubiera sido subirme al tren del software propietario, firmar acuerdos de confidencialidad y prometer no ayudar a mis compañeros..."

"...Decidí estudiar la manera en que un programador podría hacer algo por el bien común. Me pregunté si podía escribir uno o varios programas que permitiesen resucitar nuevamente a nuestra extinta comunidad. La respuesta era obvia: la primera cosa necesaria era crear un sistema operativo, el software crucial para empezar a utilizar un ordenador..."

"...Opté por crear un sistema compatible con Unix para dotarle así de portabilidad y facilitar el cambio a los usuarios de Unix. El nombre de GNU fue elegido según una tradición de los hackers, como un acrónimo recursivo de «GNU's Not Unix» ..."

"...Libre en su acepción de libertad"

A veces se malinterpreta el término de «software libre». Para empezar, no tiene ninguna relación con el precio. Lo que nos interesa es la libertad.

He aquí la definición de software libre. Un programa es software libre siempre que, como usuario particular, tengas:

- *La libertad de ejecutar el programa sea cual sea el propósito.*
- *La libertad de modificar el programa para ajustarlo a tus necesidades. (Para que se trate de una libertad efectiva en la práctica, deberás tener acceso al código fuente, dado que sin él la tarea de incorporar cambios en un programa es extremadamente difícil).*
- *La libertad de redistribuir copias, ya sea de forma gratuita, ya sea a cambio del pago de un precio.*
- *La libertad de distribuir versiones modificadas del programa, de tal forma que la comunidad pueda aprovechar las mejoras introducidas..."*

Algo interesante que surge de esta definición es que **Software Libre NO SIGNIFICA Software Gratuito.**

Al respecto dice Stallman:

"...Dado que nos referimos a la libertad y no al precio, no existe contradicción alguna entre la venta de copias y el software libre. De hecho, la libertad para vender copias es crucial: las colecciones de software libre a la venta en formato de CD-ROM son muy importantes para la comunidad y venderlas es una forma de recaudar fondos para el desarrollo de software libre..."

El desarrollo de un sistema operativo de principio a fin es un proyecto colosal, ya que tiene muchos componentes, por lo que Stallman decidió adaptar e incorporar algunas piezas existentes de Software Libre, desarrollando otras partes, como el editor GNU Emacs hacia fines de 1984. Este editor Emacs se transformó, justamente, en una de las piezas de Software Libre que comenzó a vender, a unos 150 dólares la copia.

Dice:

"...Así comenzó mi empresa de distribución de software libre, precursora de las empresas que hoy distribuyen sistemas Linux basados en GNU..."

En otro párrafo, refiriéndose a la posibilidad de transformar Software Libre en Propietario:

"... ¿Un programa es libre para cualquier usuario?

Cuando un programa de software libre deja de estar en manos de su autor, esto no significa necesariamente que siga siendo software libre para cualquiera que se haga con una copia de él. Por ejemplo, el software de dominio público, software sin copyright, es software libre, pero cualquiera puede modificarlo y hacer una versión propietaria a partir de él. Lo mismo ocurre con muchos programas libres con copyright que se distribuyen con licencias simples muy permisivas que autorizan el desarrollo de versiones propietarias modificadas..."

"...El Copyleft y la GNU GPL

El objetivo de GNU era proporcionar libertad a los usuarios, no simplemente ser popular. De modo que necesitábamos idear unos términos de distribución que impidieran que el software de GNU se convirtiera en software propietario. El método que empleamos se denominó copyleft.

Copyleft utiliza la ley de copyright, pero dándole la vuelta para servir a un propósito opuesto al habitual: en lugar de privatizar el software, ayuda a preservarlo como software libre.

La idea fundamental del copyleft es que se autoriza la ejecución del programa, su copia, modificación y distribución de versiones modificadas, siempre que no se añada ninguna clase de restricción a posteriori. De este modo, las libertades cruciales que definen el «software libre» quedan garantizadas para cualquiera que posea una copia; estas libertades se convierten en derechos inalienables.

Para que el copyleft sea efectivo, las versiones modificadas deberán ser libres también. Esto garantiza que cualquier tarea basada en nuestro trabajo se pondrá a disposición de la comunidad si llegara a publicarse. Cuando los programadores que tienen empleo se ofrecen voluntariamente a mejorar el software GNU, sólo el copyleft impide que sus jefes les digan: «No podéis compartir esos cambios, porque vamos a utilizarlos para crear nuestra versión propietaria del programa»...

“...La Free Software Foundation

*A medida que aumentaba el interés por Emacs, otros vinieron a sumarse al proyecto GNU, y decidimos que era el momento de volver a buscar fuentes de financiación. De este modo, en 1985 creamos la **Free Software Foundation**, una organización sin ánimo de lucro dedicada al desarrollo de software libre. La **FSF** también se hizo con la empresa de distribución de copias de Emacs, a lo que más tarde añadiría otros programas libres, no sólo de GNU, así como la venta de manuales libres...”*

“...El GNU Hurd

En 1990, el sistema GNU estaba casi terminado. Faltaba crear un solo componente central, el kernel...”

El Kernel, como su nombre lo indica, es el núcleo del sistema operativo y en el proyecto GNU se lo denominaba “Hurd”.

“...La consolidación del Hurd ha llevado muchísimos años...”

“...Linux y GNU/Linux

El GNU Hurd no estaba listo para producción. Afortunadamente, otro kernel estaba a nuestra disposición. En 1991, Linus Torvalds (finlandés) desarrolló un kernel compatible con Unix y lo llamó Linux. En el año 1992, la combinación de Linux con el incompleto sistema GNU dio como resultado un sistema operativo libre. (Esta combinación fue, por supuesto, una labor extraordinaria). Gracias a Linux podemos ejecutar hoy una versión del sistema GNU. Denominamos esta versión GNU/Linux para explicar su composición, una combinación del sistema GNU con Linux como kernel...”

“...«Open Source» (código fuente abierto)

La tarea de enseñar a los nuevos usuarios el valor de la libertad se complicó de forma acusada en 1998, cuando parte de la comunidad decidió abandonar el término «software libre» y empezó a hablar de «software de código abierto».

Los partidarios de este término trataban de evitar la confusión entre «libre» y «gratis», un objetivo muy legítimo. Pero otros intentaban dejar a un lado los principios que habían impulsado la creación del software libre y el proyecto GNU, procurando así atraer a los ejecutivos y a los usuarios de empresas, quienes comparten mayoritariamente una ideología que antepone las ganancias económicas a la libertad, a la comunidad, a los principios. De modo que la retórica del «código abierto» se concentra en la posibilidad de crear un software de alta calidad y capacidad, pero rehuye las nociones de libertad, comunidad y principios..."

Lo que nos está diciendo Stallman en este último párrafo, es que “Open Source” NO ES LO MISMO QUE SOFTWARE LIBRE.

Y continua:

"...Por qué «software libre» es mejor que «open source»

DADO QUE EL SOFTWARE LIBRE te daría la misma libertad con cualquier otro nombre, qué nombre usemos marca una gran diferencia: palabras distintas transmiten distintas ideas..."

"...El término open source se asoció rápidamente con un enfoque distinto, una filosofía distinta, e incluso diferentes criterios para decidir qué licencias son aceptables.

El movimiento de software libre y el movimiento open source son hoy en día movimientos separados con diferentes puntos de vista y objetivos, aunque podamos y trabajemos juntos en algunos proyectos prácticos.

La diferencia fundamental entre los dos movimientos está en sus valores, en su visión del mundo. Para el movimiento open source, la cuestión de si el software debe ser de fuente abierta es una cuestión práctica, no ética. Como lo expresó alguien, «el open source es un método de desarrollo; el software libre es un movimiento social».

Para el movimiento open source, el software no libre es una solución ineficiente. Para el movimiento de software libre, el software no libre es un problema social y el software libre es la solución..."

"...El movimiento del software libre y el movimiento open source son como dos campos políticos dentro de la comunidad del software libre..."

"...Estamos en desacuerdo en los principios básicos, pero estamos mas o menos de acuerdo en las recomendaciones prácticas. Así que trabajamos juntos en muchos proyectos específicos. No pensamos en el movimiento open source como enemigo. El enemigo es el software propietario..."

Según Wikipedia:

La filosofía del Open Source centra su atención en la premisa de que al compartir el código, el programa resultante tiende a ser de calidad superior al software propietario, es una visión meramente técnica. Por otro lado, el Software Libre funciona bajo el ideal que el software propietario, al no poder compartirse es antiético, dado que prohibir compartir entre seres humanos va en contra las leyes naturales.

El movimiento Open Source tiene un decálogo que debe cumplir un código para poder llamarse "Open Source" (es de hacer notar que estas 10 premisas son completamente equivalentes con las 4 libertades o principios del Software Libre), éstas son :

- 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.*

Es en este punto 9, creo yo, donde se marca una diferencia importante con el movimiento de Software Libre, al permitir "mezclar" software de código abierto con software propietario. Al respecto dice Stallman:

"...En el encuentro del «Día de fabricantes de open source» en agosto de 1998, varios de los fabricantes comerciales invitados dijeron que ellos solamente tenían en mente convertir una parte de su trabajo en software libre —u open source. El enfoque de sus negocios está en desarrollar accesorios propietarios —software o manuales— para vender a los usuarios de software libre. Nos piden que consideremos esto como legítimo, como parte de nuestra comunidad, debido a que parte del dinero se dona al desarrollo de software libre..."

- 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.*

El Software Libre en la Universidad

A este respecto Stallman recomienda:

"...Cómo promover el software libre si trabajas en la universidad

EN EL MOVIMIENTO DEL SOFTWARE LIBRE creemos que los usuarios de ordenadores deberían tener libertad para cambiar y redistribuir el software que utilizan. El adjetivo «libre» en el software libre hace referencia a la libertad: libertad del usuario para ejecutar, modificar y redistribuir software. El software libre contribuye al saber humano, al contrario que el software propietario. Por este motivo, las universidades deberían fomentar el software libre, para hacer una aportación al progreso del conocimiento humano, del mismo modo que deben animar a científicos y académicos a publicar sus obras.

Pero el software (y la ciencia) despiertan la codicia en un gran número de gerentes universitarios: consideran los programas como una potencial fuente de ingresos y no como aportaciones al saber humano. Los programadores de software libre llevan conviviendo con esta tendencia desde hace casi veinte años.

Cuando comencé a desarrollar el sistema operativo GNU en 1984, lo primero que hice fue renunciar a mi trabajo en el MIT. Hice esto precisamente para que así la oficina de licencias del MIT fuera incapaz de interferir en la publicación de GNU como software libre.

Había diseñado una estrategia para licenciar los programas contenidos en GNU que garantizaría que todas las versiones modificadas seguirían siendo software libre, una estrategia que acabaría convirtiéndose en la GNU General Public License (GNU GPL). (de la cual vamos a hablar en un ratito) No quería tener que rogarle a la administración del MIT que me permitiera usarla.

Con el paso de los años, varias filiales universitarias han acudido con frecuencia a la Free Software Foundation para asesorarse sobre la forma de negociar con los gerentes que opinan que el software es tan sólo algo que vender. Un buen método, aplicable incluso a proyectos específicamente financiados, consiste en basar su trabajo en un programa ya existente publicado con GNU GPL. De esta forma, se puede responder a los gerentes: «No podemos publicar la versión modificada, a menos que sea con GNU GPL, de otro modo estaríamos infringiendo el copyright». Una vez desaparecido cualquier rastro del símbolo del dólar en sus ojos, por lo general consentirán en publicarlo como software libre..."

Los tipos de licencias usadas en el proyecto GNU

Se utilizan tres tipos de licencias:

1. Licencia Pública General GNU [GNU-GPL]
2. Licencia Pública General Menor GNU [GNU-LGPL]
3. Licencia de Documentación Libre GNU [GNU-FDL]

Según Wikipedia:

La GNU GPL (General Public License o licencia pública general) es una licencia que obliga a los licenciarios a propagar ciertos derechos y libertades en relación al software sobre el que la licencia se aplica. La restricción básica que manda esta licencia es la obligatoriedad de poner a disposición los fuentes de un programa, entendiendo fuentes como la forma en la que es preferible crear modificaciones, a todas aquellas partes que reciban una versión compilada. Por este mecanismo de obligaciones "transitivas", la licencia GNU GPL es caracterizada como una licencia "viral". Este tipo de licencias que se centra en extender a todos permisos o libertades sobre un programa es conocido como copyleft.

Esta licencia fue creada por la Free Software Foundation a mediados de los 80 como parte del ideario de esta organización de propugnar el software libre.

La licencia sólo habla de la distribución de software y nunca del uso. Por lo tanto es completamente posible utilizar versiones modificadas de un programa sin que se imponga la necesidad de compartir de ninguna manera esas modificaciones

La GNU LGPL (Licencia Pública General Menor), se aplica a ciertas bibliotecas y es bastante diferente de la Licencia Pública General ordinaria.

Se usa esta licencia para ciertas bibliotecas con el objeto de permitir el enlace de las mismas dentro de programas no libres.

La Licencia Pública General Menor introduce un criterio más laxo para el enlace de otro código con la biblioteca.

Se denomina de esta manera porque hace «menos» que la Licencia Pública General ordinaria para proteger las libertades del usuario.

Sin embargo, la licencia Menor proporciona ventajas en circunstancias especiales.

Por ejemplo, en raras ocasiones puede haber una especial necesidad de fomentar, al máximo posible, el uso de una biblioteca determinada, de forma que esta se convierta en un estándar. Para ello, se debe permitir a los programas no libres el uso de estas bibliotecas. Un caso más frecuente es el de una biblioteca libre que desempeñe la misma función que la que realizan las bibliotecas no libres más utilizadas. En este caso, hay poco que ganar limitando la biblioteca únicamente al software libre, de manera que se usa la Licencia Pública General Menor.

En otros casos, el permiso para usar una biblioteca determinada en programas no libres aumenta el número de usuarios de gran cantidad de software libre. Por ejemplo, el permiso para utilizar la biblioteca GNU C en programas no libres posibilita a mucha más gente utilizar el sistema operativo GNU, así como su variante, el sistema operativo GNU/LINUX.

Según Wikipedia:

La GNU FDL (Free Documentation Licence o Licencia de Documentación Libre) es una licencia copyleft para contenido libre, diseñada por la Fundación del Software Libre (FSF) para el proyecto GNU.

La licencia fue diseñada para documentación de software y otras referencias y materiales institucionales. Estipula que cualquier copia del material, incluso de ser modificado, debe llevar la misma licencia. Dichas copias pueden ser vendidas pero, de ser producidas en cantidad, deben ser distribuidas en un formato que garantice futuras ediciones. Wikipedia es el mayor proyecto que utiliza esta licencia.

Muchas personas y grupos consideran a GFDL como una licencia no libre, debido en parte al uso de texto "invariable" que no puede ser modificado o eliminado, lo cual afecta también algunos usos válidos.

El interés de este debate para los objetivos del congreso

En forma genérica podemos decir que el congreso propicia la discusión de los diferentes temas sociales y tecnológicos que pueden apuntar desde lo educativo, y proyectándose al conjunto de la sociedad, para un proceso de integración regional latinoamericana y crecimiento de la soberanía. Desde el punto de vista del Software Libre, porque creemos que un proceso cooperativo de los países de la región, en este sentido, puede ayudar a reforzar la soberanía. De gran importancia para ir en ese camino es el estado de la legislación en diferentes países latinoamericanos. En el documento del *Plan de Trabajo* de esta mesa encontrarán algunas referencias.

Esta tendencia a legislar sobre el uso del Software Libre, sobre todo en organismos públicos, muestra claramente que el debate NO ES solo tecnológico, sino que entra en el campo político, ya que como diría Stallman, el software libre es un movimiento social, que se concentra en las nociones de libertad, comunidad y principios...

Veamos esto a la luz de los objetivos del congreso:

Los objetivos del congreso

Hay cuatro objetivos que me interesa destacar

Objetivos generales:

2. Articular propuestas en pos de la construcción de un Modelo de Educación Superior Nacional, Popular y Latinoamericanista, atento a dar respuesta a las necesidades de desarrollo de nuestros pueblos latinoamericanos
4. Construir una plataforma educativa, cultural, democrática y de gestión participativa-popular y de compromiso social que sostenga el ALBA como proyecto estratégico de los Pueblos de América Latina y Caribe.

Objetivos del Bloque Temático 4: La integración Estratégica para nuestros Pueblos

3. Avanzar propositivamente en los ejes estratégicos de desarrollo de un proyecto nacional-latinoamericano que aporten a la transformación en la educación superior y social.
4. Trazar ejes estratégicos orientadores para desarrollar nuevos contenidos en los planes de estudio que apunten a una visión de un Proyecto de Universidad comprometido con los cambios sociales nacionales-latinoamericanos.

Nos planteamos que el crecimiento del uso de Software Libre en Latinoamérica puede ayudar a reforzar la soberanía, permitiendo una mayor independencia de los monopolios. Debemos impulsar ese crecimiento a través de un proceso cooperativo de los países de la región. O como dicen los alumnos de UTN Córdoba, en su trabajo sobre el "Cliente libre de Mensajería":

"...La participación de programadores latinoamericanos en el desarrollo de proyectos de Software Libre resultará en más Software Libre para todos..."

La Universidad, formadora de profesionales, es un partícipe necesario en este proceso. Entonces, las universidades latinoamericanas tienen que:

- Transformar ese "Modelo de Educación Superior"
- "Construir una plataforma educativa"

- "Avanzar en propuestas que aporten a la transformación en la educación superior"
- "Trazar ejes estratégicos para desarrollar nuevos contenidos en los planes de estudio"

Para que apunten a la formación de profesionales concientes de la libertad, y en el área de Sistemas, propicie el uso de Software Libre, que en definitiva es un movimiento social, que se concentra en las nociones de libertad, comunidad y principios...

Material de trabajo para el Debate

Los objetivos del congreso

Objetivos generales:

1. Debater el rol de la Universidad Pública en el actual proceso de cambio social y de integración regional Latinoamericana.
2. Articular propuestas en pos de la construcción de un Modelo de Educación Superior Nacional, Popular y Latinoamericanista, atento a dar respuesta a las necesidades de desarrollo de nuestros pueblos latinoamericanos
3. Impulsar una Reforma Universitaria en Argentina y Latinoamérica con el fin de instituir a la Universidad Pública en plataforma fundamental del desarrollo nacional y de la integración cultural, científica y social de los pueblos latinoamericanos.
4. Construir una plataforma educativa, cultural, democrática y de gestión participativa-popular y de compromiso social que sostenga el ALBA como proyecto estratégico de los Pueblos de América Latina y Caribe.

Objetivos del Bloque Temático 4: La integración Estratégica para nuestros Pueblos

5. Profundizar la integración latinoamericana como proyecto estratégico que nos permita abordar los grandes problemas nacional-latinoamericanos y las soluciones desde los Pueblos.
6. Generar un hecho concreto de integración donde la Universidad no esté aislada de los problemas nacional-latinoamericanos, y sea una herramienta de liberación social.
7. Avanzar propositivamente en los ejes estratégicos de desarrollo de un proyecto nacional-latinoamericano que aporten a la transformación en la educación superior y social.
8. Trazar ejes estratégicos orientadores para desarrollar nuevos contenidos en los planes de estudio que apunten a una visión de un Proyecto de Universidad comprometido con los cambios sociales nacionales-latinoamericanos.

Guía de preguntas para las mesas de trabajo del Bloque Temático 4

1. Estado de situación del Área temática de esta Mesa
2. Correlación de fuerzas nacionales, latinoamericanas y generales para sostener un proyecto Soberano
3. La necesidad de un proyecto de Soberanía
4. Los Actores socio-económicos que pueden sostener y potenciar el Proyecto Soberano:
 - a. Empresas estatales:
 - b. Pequeñas empresarios
 - c. Profesionales
 - d. Obreros
5. La Educación Superior que necesitamos para un proyecto soberano Nacional latinoamericano:
 - a. La universidad Nacional y la educación superior para la soberanía e independencia
 - b. El Movimiento Universitario, su proyecto, su programa y sus tareas

6. El proyecto de unidad e integración latinoamericana como visión y territorio de desarrollo de la soberanía –eje temático-:
 - a. Obstáculos, amenazas
 - b. Apoyos,
7. La Ciencia y la Tecnología para el desarrollo de la Soberanía eje temático:
 - a. Las empresas que producen tecnología para la Energía Soberana
 - b. Las empresas subsidiarias para la energía soberana
 - c. Los profesionales para la soberanía energética
 - d. Los obreros para la soberanía energética
 - e. La educación, la investigación, la Ciencia y la tecnología para soberanía energética
8. La organización social y política de la sociedad y de los Pueblos para sostener y desarrollar el proyecto

Estado de la legislación en algunos países latinoamericanos:

Venezuela: Decreto N° 3.390 Fecha: 23 de diciembre de 2004. Dice, entre otras cosas:

“...La Administración Pública Nacional empleará prioritariamente Software Libre desarrollado con Estándares Abiertos, en sus sistemas, proyectos y servicios informáticos. A tales fines, todos los órganos y entes de la Administración Pública Nacional iniciarán los procesos de migración gradual y progresiva de éstos hacia el Software Libre desarrollado con Estándares Abiertos...”

Perú: PROYECTO DE LEY DE USO DE SOFTWARE LIBRE EN LA ADMINISTRACIÓN PÚBLICA de Los Congresistas de la República, EDGAR VILLANUEVA NÚÑEZ y JACQUES RODRICH ACKERMAN, del 09 de abril de 2002. Dice:

“...Empléase en todas las instituciones del Estado el uso exclusivo de programas o software libres en sus sistemas y equipamientos de informática...”

Brasil: El Estado Brasileño de Rio Grande do Sul legisla la adopción de Software Libre en la Administración Pública, mediante la ley Nro. 11871 [LEY] aprobada el 19 de diciembre del 2002. La ley establece que la administración pública debe usar preferentemente "software que sea abierto, libre de restricciones propietarias para su cesión, modificación y distribución".

“Art. 1º A administração pública direta, indireta, autárquica e fundacional do Rio Grande do Sul, assim como os órgãos autônomos e empresas sob controle do Estado utilizarão preferencialmente em seus sistemas e equipamentos de informática programas abertos, livres, de restrições proprietárias quanto a sua cessão, alteração e distribuição.”

La ley está siendo desafiada en la corte como inconstitucional, bajo el caso número 3059/03. Los argumentos presentados en el inicio [INI] son que el estado no puede regular las ofertas, ya que este tipo de regulaciones está reservada a la Unión, que la ley falla en respetar el principio de isonomía (igual distribución de derechos y privilegios) entre los licitantes, y que la adopción de Software Libre infringiría los principios de eficiencia y economía, y que una ley de este tipo sólo podría ser propuesta por el Poder Ejecutivo, nunca por el legislativo.

Aún cuando el Consejo legal de la Unión indicó que todos estos argumentos eran falsos [AGU], la Corte Federal, donde se llevan adelante los casos, otorgó un interdicto primario y la ley está actualmente suspendida.

Argentina:

1. Nivel Nacional: Hay dos proyectos de Ley:

a. Proyecto de Ley 904-D-02 (Dragan/Becerra/Bertone) Política de utilización de software libre por el Estado Nacional:

“...Artículo 2º.- Los Poderes Ejecutivo, Legislativo y Judicial, los Organismos Descentralizados y las Empresas donde el Estado Nacional posea mayoría accionaria, emplearán en sus sistemas y equipamientos de informática exclusivamente programas (software) libres...”

b. Proyecto de Ley 6875-D-02 (Fontdevila) RÉGIMEN DE POLÍTICAS DE CONTRATACIÓN DE SOFTWARE EN EL SECTOR PUBLICO NACIONAL:

“...Dando preferencia a la contratación de programas de código abierto se favorece la independencia de manejo y control de los propios datos por parte del Estado, haciendo que la utilización y el mantenimiento del software no dependan de la buena voluntad de los proveedores, ni de las condiciones impuestas por éstos...”

En una Carta de la FVL (Fundación Vía Libre), a los integrantes de las comisiones de la Cámara de Diputados, del 19 de marzo de 2003, dice:

“...Habiendo tomado conocimiento del proyecto de ley con expediente 6875-D-02, impulsado por el Presidente de la Comisión de Comunicaciones e Informática de la Honorable Cámara de Diputados, diputado Fontdevila, aprovechamos la oportunidad para dirigirnos a los Sres. Legisladores con el fin de comunicarles las razones por las que estimamos conveniente desestimar dicho proyecto, y en cambio propiciar el tratamiento y la aprobación del proyecto 904-D-02 presentado por el diputado Dragan y otros...”

2. Provincia de Santa Fe: Ley N° 12.360 Aprobada en la Legislatura de la Provincia de Santa Fe, Argentina, el 18 de noviembre de 2004.

“...Artículo 2º: Los Poderes Ejecutivo, Legislativo, y Judicial, los organismos descentralizados y las empresas donde el Estado Provincial posea mayoría accionaria emplearán en sus sistemas y equipamientos de informática preferentemente software libre...”

3. Ciudad Autónoma de Buenos Aires: PROYECTO DE LEY 2801-D-04 REGIMEN DE POLÍTICAS DE SOFTWARE PARA EL GOBIERNO DE LA CIUDAD AUTONOMA DE BUENOS AIRES.:

“...Artículo 3º: . El sector Público de la Ciudad de Buenos Aires , el cual comprende la Administración Central (Poderes Legislativo, Ejecutivo y Judicial), las Comunas, los Organismos Descentralizados, Entidades Autárquicas, Organismos de la Seguridad Social, las Empresas y Sociedades del Estado, Sociedades Económicas con participación Estatal Mayoritaria, Sociedades de Economía Mixta y todas aquellas otras Organizaciones Empresariales donde el Estado de la Ciudad tenga participación mayoritaria en el capital, deben emplear en sus sistemas y equipamientos de informática, exclusivamente programas libres, programas de código abierto y programas de dominio publico...”