



**Universidad de las Ciencias Informáticas
Facultad 10**

**Título:
Modelos de Negocio para Comercializar Software Libre.**

**Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas**

**Autores: Acela Delgado Castillo
Ernesto Fernández Aguilar**

Tutor: MSc. Nelson W. Ferrer Martín

Consultor: MSc. Tomás López Jiménez

Ciudad de la Habana

Mayo 2009

“Año 50 de la Revolución”



“...el software libre no es una cuestión de precio, un precio bajo no resulta más libre, ni más próximo a lo libre. De modo que si redistribuyes copias de software libre, también puedes fijar un precio cuantioso para hacer algo de dinero. Redistribuir software es una actividad buena y legítima; si lo haces, también puedes sacar una ganancia por ello.”¹

Richard Stallman, 2004

¹ Software Libre para una sociedad libre, Noviembre del 2004

A mis padres con todo mi corazón, por haber dedicado toda su vida a mi formación, por confiar completamente en mí, y no haber perdido la esperanza de ver realizado este gran sueño que es de todos...

Acela

A mi familia que me ha apoyado en todos los momentos buenos y difíciles de mi vida, ellos que lo han dado todo porque este sueño se haya hecho realidad.

Ernesto

De Acela:

Quisiera agradecer en este momento, a muchas personas que de una forma u otra han hecho posible que este sueño se haga realidad, a aquellas que han confiado en todo momento en mí y me han apoyado y ayudado en el desarrollo de este trabajo, de forma especial:

- Ramona Castillo Núñez, mi mamá por brindarme su amor y apoyo incondicional.
- Fernando Delgado O’Farril, mi papá por apoyarme tanto.
- A mi novio Jairon Redondo Álvarez por compartir conmigo todo este tiempo.
- Mi profundo agradecimiento a Nelson Ferrer Martín que como mi tutor fue especial, por su apoyo incansable en todo momento.
- Agradecer al MSc. Tomás López Jiménez por su apoyo y por aportar a esta causa.
- Agradecer por su aporte a la investigación a: Ing. Daysi Oropesa Méndez de ALBET, Ing. Martha Abreu de SOFTEL, Ing. Abel Meneses Abad del Proyecto UICORNIOS, Ing. Adrián Cid Almaguer del Proyecto Archivo Nacional de Venezuela, Ing. Yordán Vélez del Centro UCID, Ing. José Ramón Hermosilla del Proyecto Filpacon, Ing. Yadier Pascual Riverón, Ing. Laya del Carmen, Dr. Pedro Y. Piñero del CENTALAD, MSc. Filiberto López Cosío y Ing. Juan A. Fung Goizueta del Proyecto SCADA; todos son de la UCI.
- Agradecer la colaboración de María Isabel Morejón de la empresa DESOFT y al MSc. Gilberto Castillo Martínez de ETECSA.
- Agradecer al Ing. Yaciel E. Tellez Toledo por dedicar varias horas en la revisión del Trabajo.
- Agradecer a mis amigos Dayaisis B. Bernis, Yanicet Aveleira y a Ernesto Fernández Aguilar por su apoyo infinito en esta etapa.
- A todos mis profesores que de una u otra forma aportaron su granito de arena a la preparación que he adquirido a lo largo de estos años de estudiante.
- A todos mis compañeros de aula que transitaron conmigo estos maravillosos años en esta honrosa misión en la cual alcanzamos nuestros sueños.
- En fin, para no olvidar a nadie, a todos aquellos que nos han apoyado durante este largo andar, a todos ustedes....,

Muchísimas Gracias de todo corazón.

De Ernesto:

A lo largo de nuestro trabajo hemos encontrado bastante apoyo de diferentes formas, es por ello que quiero agradecer:

- A nuestro Tutor: Nelson Walfrido Ferrer por su confianza, por soportarnos, por sus valiosas ideas, su disposición de atendernos, su continuo asesoramiento y su ayuda para el desarrollo de este trabajo.
- A los profesores Yoandy Lichilin, Sonia Guerrero, Yanicet Aveleira, por estar siempre dispuestos a tendernos la mano, gracias por toda su cooperación.
- A mis padres, mi hermano y mi familia en general por todo el soporte emocional que han brindado. En especial a mi tío Alberto Medina quien ha sido un pilar muy importante y debido a su experiencia y sus consejos, hemos sabido sacar provecho de nuestros propios esfuerzos y habilidades.
- Y a todas las personas que de una forma u otra han brindado su ayuda y colaboración para la realización de este trabajo.

DECLARACIÓN DE LA AUTORÍA

Declaramos ser autores del presente Trabajo de Diploma y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Acela Delgado Castillo
Autor

Ernesto Fernández Aguilar
Autor

MSc. Nelson W. Ferrer Martín
Tutor

RESUMEN

El Modelo de Negocio es el mecanismo por el cual un negocio trata de generar ingresos y beneficios. Es un resumen, de cómo una compañía planifica servir a sus clientes, que implica el concepto de estrategia y el de implementación. Para poder lograr la eficiencia de este, se analizó: ¿qué es el Software Libre?, si se puede comercializar y cómo para poder aplicarlo, entre otros conceptos generales, características, funciones y herramientas que son imprescindibles a la hora de confeccionarlo; cuyo formato puede ser aplicable a cualquier modelo que usted desee usar, para comercializar cualquier producto y/o servicio de software libre, creado en la Universidad de las Ciencias Informáticas y que pudiera ser de utilidad para alguna empresa del país que necesite vender su producto y/o servicio. Gracias a la importancia y a las ventajas que este brinda, este Trabajo de Diploma desarrolló tres propuestas principales de Modelos de Negocio aplicados a tres Casos de Estudio. El primero se llama Comercialización de servicios integrales (se basa en comercializar todo tipo de servicio que no esté asociado a ningún producto en específico, es general) y se aplicó al proyecto UNICORNIOS, el segundo: Comercialización de un producto y sus servicios asociados (se basa en comercializar un producto y ofrecer servicios asociados a este) y fue aplicado a FILPACON, y el tercero: Elaboración de sistemas a pedidos (consiste en desarrollar sistemas por medio de un contrato) se le aplicó a SCADA para PDVSA. Estos modelos, en sí, organizan y presentan los detalles importantes de los problemas reales que se vinculan con el sistema informático a construir y mantienen los aspectos éticos, técnicos y comerciales que tiene la empresa, el proyecto o la entidad que lo desarrolla.

Palabras claves: Software, Software libre, Modelos de Negocio, Modelos de Negocio para SWL.

ABSTRACT

The Business Model is a strategy by which a business attempts to generate incomes and benefits. It is a summary of how the company designs the services addressed to its clients, which implies the concept of strategy and implementation. The analysis of ¿what is Free Software? was useful to achieve its efficiency, also if this type of software can be for sale and how it can be applied, among other general concepts, features, functions and tools; these are essential matters to have in mind when we focus on make it up. This format may be applied to any model you wish to use, for selling any kind of product and/or services of free software created in the University of Information Sciences and that could be useful for some national enterprise which wants to sell its product. Due to the significance and advantages it offers, this work accomplished three main proposal of Business Model applied to three Study Cases. The first one it's named Integral Services Merchandising (based on commercializing every kind of service that it's not associated to any product specifically, it's general) applied to UNICORNIOS project, the second one: Merchandising of a Product and its Services Associated (based on commercializing a product and to offer services associated a this) applied to FILPACON, and the last one: Upon Request Systems Elaboration (it develops systems by means of a contract) applied to SCADA toward PDVSA. These models organize and show the significant details of the existing difficulties linked to the information system for building and maintaining the ethical, technical and commercial issues that the company, project or development entity deals with.

Keywords: Software, Free Software, Business Model, Business Model for Free Software.

TABLA DE CONTENIDO

INTRODUCCIÓN.....4

CAPÍTULO 1. EL SWL Y SUS MODELOS DE NEGOCIO9

1.1 Introducción 9

1.2 El Software (SW) 10

1.3 El Software Libre 10

1.3.1 El inicio 10

1.3.2 Definición del SWL 14

1.3.3 Legalidad 14

1.3.3.1 Tipos de licencias de software libre 15

1.3.3.2 Explotación de los derechos de autor y la propiedad intelectual del SWL. 21

1.4 Diferencias entre SWL y Software propietario. 22

1.4.1 Software propietario, privativo o cerrado 24

1.4.2 Software de Código Abierto (SCA) 24

1.5 Modelos de Negocio para la Comercialización de SWL. Tipos identificados en la búsqueda bibliográfica. 27

1.5.1 El Modelo de Negocio 27

1.5.2 Diferencias entre los Modelos de Negocio para SW propietario y SWL 30

1.5.3 Diferentes Modelos de Negocio para SWL 32

1.5.4 Otras clasificaciones de Modelos de Negocio sobre SWL. 38

1.6 Importancia de comercializar SWL en el mercado. 42

1.7 Aspectos económicos, académicos y éticos que se deben tener en cuenta. 43

1.8 Conclusiones 45

CAPÍTULO 2: EL SWL EN AMÉRICA LATINA. LA UCI COMO CAMPO DE ACCIÓN.46

2.1 Países de América Latina vinculados al SWL 46

2.1.1 Caso Venezuela con Decreto sobre SWL. Incremento del nivel de actividad en este campo 47

2.1.2	Brasil, Argentina y México	48
2.1.3	Significación del SWL en Cuba	50
2.2	Desarrollo del SWL en la UCI	52
2.2.1	Características y posibilidades de elaboración de SWL en la UCI	53
2.2.2	Disponibilidad actual de productos y servicios. Experiencia	54
2.2.3	Formas de comercialización de SWL utilizadas hasta ahora. Ventajas y limitaciones.	61
2.2.4	Importancia del tema para la UCI por promover los desarrollos en SWL	67
2.3	Conclusiones	68

CAPÍTULO 3. TRES PROPUESTAS PRINCIPALES DE MODELOS DE NEGOCIO PARA EL SWL A UTILIZAR EN LA UCI.....70

3.1	El punto de partida	70
3.2	Formato de Modelo de Negocio	75
3.3	Tres propuestas principales aplicadas a tres Casos de Estudio	78
3.3.1	UNICORNIOS	79
3.3.2	FILPACON	84
3.3.3	SCADA para PDVSA	88
3.4	Conclusiones parciales	94

CONCLUSIONES GENERALES95

RECOMENDACIONES.....96

BIBLIOGRAFÍA.....97

GLOSARIO DE TÉRMINOS Y SIGLAS.....113

APÉNDICES.....117

ANEXOS139

Anexo 1:	Fichas de Modelos de Negocio	139
Anexo 2:	Modelos segun IT Managers Journal.	158
Anexo 3:	Algunos proyectos de SWL que existen actualmente en la UCI.	160
Anexo 4:	La GPL y su compatibilidad entre licencias	163

Anexo 5: Gráfica sobre porcentaje de utilización de los servicios ofrecidos por los Proyectos tomados como Casos de Estudio en el Capítulo 3.	169
Anexo 6: Encuesta para conocer el grado de participación que tienen los desarrolladores en las diferentes comunidades de SWL.	170

INTRODUCCIÓN

El Software Libre (SWL) está provocando una extraordinaria reacción y un polémico debate entre sus seguidores y adversarios, donde el impacto potencial en la organización de este sector genera repercusiones en la sociedad y que cada día tiene más seguidores, al que se le unen empresas que están lanzando productos que corren bajo Linux o que liberan el código fuente de alguno de sus programas, en determinados casos.

Entre los años 60 y 70 del Siglo XX, el software no era considerado un producto sino un añadido que los vendedores de los grandes computadores de la época (los mainframes) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros. Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (Grupo de Usuarios de Computadoras DEC²). A finales de los 70, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia.

Con la llegada de los años 80 la situación empezó a cambiar. Las computadoras más modernas comenzaban a utilizar sistemas operativos propietarios, forzando a los usuarios a aceptar condiciones restrictivas que impedían realizar modificaciones a dicho software.

Con este antecedente, Richard Stallman³ comenzó a trabajar en el proyecto GNU, y en 1984 fundó la Fundación de Software Libre (FSF). Stallman introdujo la definición de free software y el concepto de "copyleft", que desarrolló para otorgar libertad a los usuarios y para restringir las posibilidades de apropiación del software. Su objetivo era construir un sistema operativo completo y libre, y para esto la FSF ideó la Licencia Pública General (GPL), para asegurarse de que el software producido por GNU se mantuviera libre y para promover su producción.

Es muy importante llevar al Software Libre más allá de la comunidad técnica, perfeccionando la usabilidad de las aplicaciones y confeccionando paquetes integrados para que los usuarios no especializados se

² DEC Corporación de Equipamiento

³ Richard Stallman programador en el Laboratorio de Inteligencia Artificial del MIT

decidan a migrar. Con su adopción muchos usuarios se pueden llegar a beneficiar de programas más seguros a precios más factibles, y muchas empresas y países subdesarrollados de América Latina lograrían su soberanía tecnológica para lograr un mayor progreso en este campo. En Cuba es bueno destacar que a pesar de que se han logrado avances significativos, aún no se cuenta con la experiencia necesaria para sacar provecho de estos logros.

Actualmente, en la Universidad de las Ciencias Informáticas (UCI), existen varios proyectos dedicados al desarrollo del SWL y estos se encuentran agrupados en diferentes Polos Productivos, donde sus productos/servicios no cuentan con Modelos de Negocio definidos, que son necesarios a la hora de comercializarlos en el mercado. La mayoría de sus líderes están conscientes de que se deben utilizar estos modelos para saber como sacarle provecho a lo que ellos desean comercializar, pero no tienen conocimiento de cuáles pudieran ser aplicables en cada caso.

Es por esto, que este Trabajo de Diploma está encaminado en proponer Modelos de Negocio que vayan más allá de una prestación de servicios, tomando en consideración la idea de hacer negocio con el software libre, bajo el respeto de las cuatro libertades y las condiciones de licencia que se deben cumplir. Dicho modelo debe ser concretado por la empresa ALBET, que es la encargada de llevar a cabo todo el proceso de comercialización que se realiza en la UCI, a través de tres propuestas principales de las que se esperan resultados económicos para la Universidad, ya que esta va a elaborar principalmente su sistema en SWL, y por consiguiente alcanzar un rango considerable en el crecimiento de la economía nacional.

A raíz de la problemática antes planteada surge la siguiente **Pregunta de Investigación**: ¿Cómo contribuir a erradicar el desconocimiento existente en la UCI sobre la comercialización de productos /servicios de SWL? Esto constituye el **Problema Científico** de esta investigación.

Se puntualiza como **Objeto de Estudio**, los Modelos de Negocio para SWL y dentro de éste el **Campo de Acción**, adaptado a los Modelos de Negocio para SWL en la UCI.

Basado en la idea anterior se define como **Objetivo General** de esta investigación: Elaborar y proponer tres Modelos de Negocio para SWL, que puedan ser utilizadas en la toma de decisiones para mejorar la comercialización de productos/servicios de la UCI.

Como **Objetivos Específicos** se conformaron los siguientes:

1. Describir de forma general qué es el SWL, sus libertades y licencias.
2. Analizar el término “Modelo de Negocio” en el área del SWL y sus posibilidades de utilización por la UCI.
3. Identificar los diferentes Modelos de Negocio existentes para el SWL.
4. Diagnosticar los proyectos productivos y estructuras de comercialización de la UCI para determinar los esquemas de negocio de SWL utilizados en la Universidad.
5. Formular propuestas de Modelos de Negocio a utilizar por la UCI en los productos/servicios de SWL y aplicarlos a diferentes Casos de Estudio.

La **Idea a Defender** de esta investigación se declara a continuación: La aplicación de Modelos de Negocio, que se proponen para la UCI, puede producir avances en los procesos de toma de decisiones, en el tiempo de su aplicación en los proyectos, así como el aumento competitivo en el exterior con sus productos/servicios.

Para que el avance de este trabajo arrojara a resultados beneficiosos se definieron una serie de **Tareas de Investigación** que orientaron los esfuerzos en la investigación para darle cumplimiento a los Objetivos Específicos de esta investigación.

1. Análisis de los fundamentos teóricos de los Trabajos de Diplomas relacionados con el tema que sirvan de apoyo para el desarrollo de la investigación.
2. Extracción de la amplia red de información internacional aquellos elementos, conceptos y modelos sobre comercialización de SWL, que brinden un conocimiento preciso y coordinado como punto de partida del proceso de enseñanza-aprendizaje y guía para el trabajo de los colectivos profesionales y docentes.
3. Entrevistas al personal de empresas cubanas estrechamente ligadas a la informática y a ingenieros de la UCI, relacionados con el tema.
4. Análisis de conceptos, modelos y características de la comercialización de SWL.
5. Aplicación de tres propuestas principales de Modelos de Negocio a proyectos productivos de la UCI.

Para efectuar la investigación científica se siguió una **Estrategia de Investigación** de acuerdo a las circunstancias específicas de ésta. El tipo de estrategia seguida para su elaboración fue la **estrategia exploratoria**, de manera que fuera posible el enriquecimiento de los conocimientos referente al tema en la Universidad.

Métodos científicos utilizados en esta investigación:

Los **Métodos del Nivel Teórico** empleados para desarrollar este trabajo se encuentra el **analítico-sintético** a través del cual se analizaron varios documentos relacionados con los Modelos de Negocio y SWL, resumiendo sus características y aspectos más importantes que estas poseen. También se utilizó el **histórico – lógico** mediante el cual se realiza un estudio histórico de todo el proceso de las SWL, se ponen de manifiesto la lógica interna de su desarrollo, así como sus antecedentes, evolución y tendencias, que sirven como propuesta de Modelos de Negocio para su comercialización.

Los **Métodos del Nivel Empíricos** utilizados fueron la **observación** pues a través de esta se recopiló toda la información referente a los distintos conceptos utilizados que dan respuestas a las preguntas científicas planteadas, además se logró conocer los diferentes Modelos de Negocio que se pueden utilizar. Además, se realizaron **entrevistas** y **encuestas** a líderes de diferentes proyectos productivos de la UCI y a especialistas sobre el tema, para conocer sobre la existencia y el uso de modelos en la comercialización de sus productos/servicios. El **experimento** se aplicó en la práctica en tres propuestas principales de Modelos de Negocio, a tres proyectos productivos de la UCI, como ejemplo de estos modelos para la comercialización de SWL.

Este informe está estructurado con la respectiva secuencia; Introducción, Capítulos 1, 2 y 3, Conclusiones, Recomendaciones, Bibliografía, Siglas y Glosario de Términos, Apéndices y Anexos.

En el **Capítulo 1 – “El SWL y sus Modelos de Negocio”**, se realiza un breve resumen acerca de las características del SW en general. Se muestran los aspectos esenciales acerca del SWL, como su definición, libertades y licencias. Cuál es su filosofía, con mayor énfasis en los factores que la mueven y diferencian, principalmente en lo referente a la contribución en el desarrollo y las contradicciones con su propiedad y comercialización. Se enfatiza en el Modelo de Negocio, cuáles son los nuevos modelos y su influencia en el mercado actual que en este trabajo se exponen. De igual forma, presenta una serie de

aspectos éticos, económicos, entre otros, que se deben tener en cuenta a la hora de conformarlo. Este capítulo finaliza con una breve conclusión de lo antes analizado.

En el **Capítulo 2 – “El SWL en América Latina. La UCI como campo de acción.”**, se hace un breve análisis de cómo se manifiesta el SWL en Latinoamérica tomando como ejemplo: Brasil, Argentina, México, así como el caso Venezuela con un Decreto sobre la utilización de SWL en las entidades gubernamentales y el incremento del nivel de actividades en este campo, y más adelante se enfatiza en el auge de este tipo de software en Cuba. Luego se llega al lugar de acción donde se aplica este trabajo investigativo: la UCI. En esta sección se tratan las características y posibilidades de elaboración de SWL en la Universidad, su disponibilidad actual de productos y servicios y experiencias sobre el tema. Asimismo, las formas de comercialización de SWL utilizados hasta el momento, sus ventajas y limitaciones y la importancia que tiene comercializarlo en el exterior. El capítulo termina con una pequeña conclusión de lo antes expuesto.

En el **Capítulo 3 - “Tres Propuestas Principales de Modelos de Negocio para el SWL a utilizar en la UCI”**- después de analizar los Modelos de Negocio expuestos en el capítulo 1, las 28 fichas del Anexo 1 y las formas de comercialización existentes en la UCI, se conformaron una serie de fichas enumeradas alfabéticamente desde la A-J, que sirven como punto de partida para la aplicación de las tres propuestas principales de Modelos de Negocio para SWL, a tres proyectos productivos de la UCI. Estas propuestas siguen un mismo formato de Modelo de Negocio que es aplicable a cualquier propuesta. Luego se concluye el capítulo, de forma general.

CAPÍTULO 1. EL SWL Y SUS MODELOS DE NEGOCIO

1.1 Introducción

La evolución de la tecnología de la información ha sido muy acelerada y es por esto que la sociedad es más liberal y abierta, gracias a los cambios tecnológicos en la información y que cada día se hace más exigente donde las personas necesitan acceder y buscar libremente la información que brinda la Web y considerar un espacio para que puedan tomarla y estudiarla, esto permite que futuras generaciones aprendan sobre la necesidad de contar con métodos modernos para poder tener todo el conocimiento y los datos que se generan día tras día.

En este tema, el presente capítulo brinda una panorámica de los inicios del SWL, sus características y cuáles fueron los acontecimientos que dieron lugar al surgimiento de un movimiento más revolucionario que se basa en la solidaridad intelectual y en la entrega total, de acuerdo con el actual pensamiento de los desarrolladores. El principal objetivo es dar la libertad a los usuarios, reemplazando el software con términos de licencia restrictivos por software libre; para establecer un espacio virtual (ciberespacio) abierto con servicios a bajo precio y disponibilidad de la información para todo el mundo.

Se analiza el software libre con sus productos y servicios que, por el hecho de ser elaborado bajo código abierto, muchos creen que no es posible su comercialización. Este trabajo demuestra el desacierto de esta creencia a través de las diferentes maneras de obtener ganancias con el software libre sin restringir su código y la importancia que tiene para muchas empresas optar por él, teniendo en cuenta los aspectos éticos, económicos y legales a la hora de su comercialización.

1.2 El Software (SW)

- **Definición de Software**

“El SW es aquel conjunto de instrucciones cuya finalidad es interactuar con el hardware y realizar tareas específicas para el usuario (sistemas operativos, hojas de cálculo, gestores de bases de datos, procesadores de texto, navegadores de Internet, entre otros). Su evolución como producto e industria ha obedecido a modelos, conceptos y normas muy controvertidos, por no responder al modelo clásico industrial de la revolución que dio origen a la sociedad capitalista. Lo anterior, no hay dudas que obedece a la naturaleza de la creación intelectual, cuya industrialización es totalmente diferente a la de los productos industriales clásicos.” [1]

El software posee diversas características que le dan un toque especial a su proceso de fabricación y distribución, donde cada producto es único por sus características, pero en todos los casos hay costos, estos nunca son nulos, aunque lleguen a ser pequeños en cuanto a unitarios, pero enormes en cuanto a totales. Por otro lado, toda versión tiene un ciclo de vida, necesidad de mejoras, corrección de errores y otras necesidades de soporte y servicio que siempre generan costos cuando el modelo de negocios es serio y quiere mantener y aumentar los clientes y usuarios, por ello si no se ven estos costos y algún mecanismo del modelo que posibilite ingresos para por lo menos financiar los gastos, ese modelo de negocio fracasa o se estanca.

1.3 El Software Libre

1.3.1 El inicio

En la sección anterior se analizó al software de forma general para dar paso a un ámbito más específico como el SWL; para ello se consultaron tres etapas de su inicio, expuestas en el libro “Perspectives on Free and Open Source Software” y que se describen brevemente a continuación: **[idem, 2]**

1. Primera Era: principios de 1960 a principios de 1980.

Muchos de los aspectos importantes de los sistemas operativos de los ordenadores e Internet fueron desarrollados en ambientes académicos como Berkeley y MIT durante los años sesenta y setenta, así como en las corporaciones centrales de investigación dónde los investigadores tenían un gran convenio de autonomía. Por esos años, los programadores de diferentes organizaciones compartieron la operación básica del código fuente de los programas. Muchos de los esfuerzos de los desarrollos cooperativos en los años setenta estaban enfocados en la creación de un sistema operativo que pudiera correr en múltiples plataformas.

En 1974 se da a conocer el nuevo Sistema Operativo UNIX⁴, generándose de esta manera un gran entusiasmo en la comunidad académica, pues lo ven como una alternativa para la enseñanza. Por problemas legales a la compañía AT&T se le impide comercializarlo. Debido a esto AT&T distribuye copias a numerosas universidades y agencias gubernamentales para propósitos educativos y de investigación, incluyendo una cantidad significativa de código.

El software fue instalado de una institución a otra, para usarlo libre de cargos o bajo un pago nominal. Las demás innovaciones fueron hechas a algunos de los sitios dónde el software había sido instalado y eran compartidas con otros. El proceso de compartir el código fue avanzando gradualmente con la difusión de USENET; una red de computadora comenzada a inicios de 1980 para vincularse a la comunidad Unix.

Estos proyectos de desarrollo del software cooperativo fueron emprendidos sobre una base informal, por lo que no se hizo ningún esfuerzo por delinear los derechos de propiedad o restringir la reutilización del software. Esta informalidad pasó a ser una problemática a principios de 1980, cuando AT&T comenzó a darle fuerzas a los derechos de propiedad intelectual relacionados a UNIX.

⁴ UNIX es un sistema operativo comercial muy utilizado en entornos empresariales, a la primera versión del UNIX surgida en 1969 le llamaron UNICS.

2. Segunda Era: de 1980 a principios de 1990

Como respuesta a estas amenazas legales surgieron los primeros esfuerzos para formalizar el terreno legal sobre el proceso de desarrollo del software cooperativo. Este movimiento dio lugar a la segunda era de desarrollo de este software. Así fundó una organización en 1984 que denominó Free Software Foundation (FSF) con el fin de conseguir fondos para el desarrollo y la protección del SWL y sentó los fundamentos éticos del mismo, con documentos como “El Manifiesto GNU”.

Poco después, la FSF introdujo una importante innovación, consistía en un procedimiento de licenciamiento formal que pretendía evitar la aserción de los derechos de patente que involucraba al software cooperativo (algunos creyeron que AT&T lo había hecho en el caso de Unix). En cambio, para poder modificar y distribuir el GNU (un “acrónimo recursivo” que significa que “GNU no es UNIX”), los diseñadores del software tenían que estar de acuerdo en hacer disponible libremente el código fuente (o a un costo nominal).

Como parte de la Licencia Pública General (GPL, también conocida como “copylefting”), el usuario también tenía que estar de acuerdo en no imponer restricciones de licenciamiento a otros. Además, todas las mejoras al código habían de ser licenciados bajo las mismas condiciones. Son estas condiciones contractuales las que distinguen al software del código abierto del shareware (donde solo los archivos binarios son disponibles sin restricciones, posiblemente solo durante un período de prueba) y software del dominio público (donde ninguna restricción se impone a los usuarios del código fuente).

El Proyecto GNU también desarrolló una serie de importantes características organizativas. En particular, emplearon un modelo donde las contribuciones de muchos desarrolladores fueron aceptadas. La versión oficial del programa fue controlada por un menor subconjunto de individuos estrechamente involucrados con el proyecto, o en algunos casos, por un líder.

3. Tercera Era: principios de 1990 hasta Hoy

El avance del acceso a Internet en los primeros años de 1990 produjo una aceleración en el desarrollo del código abierto⁵. Se expandió el volumen de las contribuciones y la diversidad de los colaboradores, y surgieron numerosos proyectos de fuente abierta; el más conocido es Linux (un sistema operativo desarrollado por Linus Torvalds⁶ en 1991). Más tarde, las interacciones entre las compañías comerciales y la Comunidad de Open Source devienen conjuntamente en los años noventa.

Otra innovación durante este período fue la proliferación de una alternativa para un acercamiento al licenciamiento del software cooperativo. Durante 1980, la GPL era la licencia dominante ajustada a este software, pero esta situación cambió considerablemente durante 1990. En particular, la Debian (una organización encaminada a diseminar Linux) desarrolló en 1995 las “Normas Debian de Software Libre”. Éstas permitieron a las licencias una flexibilidad mayor en el uso del programa, incluyendo el derecho para unir al software cooperativo con código propietario. Estas provisiones se adoptaron a principios de 1997 por un número de individuos involucrados con el desarrollo del software cooperativo, y consecuentemente fueron denominadas como la “Definición de Open Source.”

Actualmente, han surgido grandes intereses en cuanto al desarrollo del Software Libre, estos involucran a programadores de diferentes latitudes y organizaciones que comparten conocimientos con el fin de desarrollar y limar detalles de programas de software, quienes han sido estimulados fundamentalmente por tres factores: la rápida difusión del Software Libre, las significativas inversiones de estos tipos de proyectos y la nueva estructura organizativa.

⁵ El movimiento del software libre hace especial énfasis en los aspectos morales o éticos del software, viendo la excelencia técnica como un producto secundario de su estándar ético. El movimiento de código abierto ve la excelencia técnica como el objetivo prioritario, siendo la compartición del código fuente un medio para dicho fin.

⁶ Linus Torvalds desarrollador del kernel (núcleo del sistema operativo) basado en Unix para computadoras con microprocesadores Intel. Actualmente es el coordinador de las versiones de Linux.

1.3.2 Definición del SWL

La Fundación de Software Libre define al Software Libre de la siguiente manera:

"Software Libre" se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. De manera más precisa, se refiere a cuatro libertades para los usuarios del software:

- 0. La libertad de ejecutar el programa, con cualquier propósito.*
- 1. La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades. El acceso al código fuente es una precondition para esto.*
- 2. La libertad de distribuir copias, con lo que puede ayudar a su vecino.*
- 3. La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. El acceso al código fuente es una precondition para esto.*

"...no hay que asociar Software Libre a software gratuito, o a precio del coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así y, aunque conserve su carácter de libre, puede ser vendido comercialmente. Análogamente, el software gratis o gratuito (denominado usualmente freeware) incluye en algunas ocasiones el código fuente; sin embargo, este tipo de software no es libre en el mismo sentido que el Software Libre, al menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa". [3]

1.3.3 Legalidad

Los temas legales en el desarrollo de proyectos de código abierto, involucran asuntos de licencias de software⁷, derechos de copia y aspectos jurídicos para comercializar SWL. En el caso de las licencias se debe revisar cuál licencia aplicar al software que se va a desarrollar. Para esto, deben revisarse las

⁷ **Licencia de software** : es el instrumento legal a través del cual el autor, titular o proveedor (licenciante) establece las condiciones y los requisitos generales bajo los que se otorga el SW al usuario (licenciario). Las licencias de SWL y de SW propietario se diferencian entre sí en la amplitud y alcance de las libertades que brindan a los usuarios. [12]

licencias libres ya existentes, la compatibilidad con otras licencias libres o propietarias, la protección de la marca y protecciones de integridad. El otro aspecto legal a revisar es respecto al derecho de copia del código generado para la comunidad. Aunque en algunas ocasiones se ignora, se recomienda la elaboración de un Acuerdo de Licencia con los Contribuyentes (CLA: Contributor License Agreement por sus siglas en inglés).

El uso de estas licencias para la comercialización del software libre brinda una serie de ventajas para el licenciante; como la posibilidad de delegación de la fabricación, el mantenimiento de la titularidad de la propiedad intelectual de la tecnología, la ampliación de las operaciones a nuevos mercados, una buena relación con sus competidores y cierto control sobre las innovaciones, así como para el licenciatarario, quien puede tener una mayor facilidad para introducirse en el mercado, un mejor acceso a los avances técnicos necesarios y la posibilidad de generar nuevos productos, servicios y oportunidades de mercado.

1.3.3.1 Tipos de licencias de software libre

Tipos de licencias de software libre⁸

La variedad de licencias libres es amplia, aunque por razones prácticas la mayoría de los proyectos utilizan un pequeño conjunto de cuatro o cinco porque no quieren o no pueden dedicar recursos a diseñar una licencia propia o porque los usuarios prefieren referirse a una licencia más utilizada en lugar de analizar licencias completas.

Las licencias de software libre se pueden dividir en dos grandes familias:

1. La primera, son las licencias robustas (o licencias copyleft) incluyen las que, al estilo de la GNU GPL, imponen condiciones en caso de que se quiera redistribuir el software, condiciones que van en la línea de forzar a que se sigan cumpliendo las condiciones de la licencia original después de la primera redistribución.

⁸ Las licencias que se describen fueron consultadas el 29 de enero del 2009 de [5] y [idem, 1] y [6]

2. La otra está compuesta por las licencias permisivas, que no imponen condiciones especiales en la segunda redistribución (esto es, que sólo especifican que el software se puede redistribuir o modificar, pero no imponen condiciones especiales si se hace lo que permite, por ejemplo, que alguien que reciba el programa pueda después redistribuirlo como software propietario).

1. Licencias robustas

Las licencias robustas, llamadas habitualmente en inglés *copyleft*⁹ tratan de garantizar las libertades que otorga el autor no sólo a quien recibe el programa directamente de él, sino también a quienes los reciben. La primera de este tipo de licencias fue la GNU GPL.

➤ Licencia Pública General de GNU (GNU-GPL)

La GPL facilita la redistribución de ambos códigos, el código fuente y el ejecutable, aunque en el caso de que se redistribuya el binario, obliga a que también se permita y garantice el acceso al código fuente. Permite realizar modificaciones sin ningún tipo de restricciones, teniendo en cuenta siempre que solo se puede integrar código licenciado bajo GPL con otro código que se encuentre bajo una licencia similar o compatible (**Ver Anexo 4**). Esta característica hace que a esta licencia se le atribuya una consecuencia recursiva por garantizar la libertad del SW al cual licencie, asegurando también que el código que sea publicado con este estado nunca pueda cambiar de condición. [*idem*, 1]

➤ Licencia LGPL

Licencia Pública General Menor (Lesser) de GNU o GNU-LGPL. Con anterioridad se conocía como la licencia para Librerías de las GPL; el nombre fue cambiado hace algún tiempo. Esta licencia es muy similar a la GPL en condiciones y restricciones, pero define una alternativa, mediante la cual permite al SWL vincularse con el propietario. El uso de la LGPL solo se recomienda para casos

⁹ *copyleft* asegura la transmisión de los efectos de la licencia original a las licencias de los programas derivados. Así preserva el carácter del SL prohibiendo que desarrollos basados en programas con licencias *copyleft* se licencien o se redistribuyan con restricciones adicionales.

específicos, por ejemplo cuando no hay otra alternativa y las librerías libres no resuelven el problema, entonces es cuando se recomienda su uso. Cuando la librería tiene funcionalidades extraordinarias respecto a las de su tipo y ofrece ventajas por sobre las demás, es preferible usar la GPL y no la LGPL. De esta forma se incentiva a los desarrolladores que quieran usar la librería a liberar también su código. **[idem, 1]**

➤ **Licencia de Sleepycat**

Es la licencia por la que la empresa Sleepycat distribuye sus programas (como el conocido Berkeley DB). Obliga a ciertas condiciones, siempre que se redistribuye el programa o trabajos derivados del programa; a ofrecer el código fuente (incluyendo las modificaciones, si se trata de un trabajo derivado) y a que la redistribución imponga al usuario las mismas condiciones. Aunque mucho más corta que la GNU GPL, es muy similar a ella en sus efectos principales. **[idem, 5]**

➤ **eCos License 2.0**

Es la licencia bajo la que se distribuye eCos, un sistema operativo de tiempo real. Es una modificación de la GNU GPL que no considera que el código que se enlace con programas protegidos por ella queden sujetos a las cláusulas de la GNU GPL si se redistribuyen. Desde este punto de vista, sus efectos son similares a los de la GNU LGPL **[idem, 5]**

➤ **Affero General Public License**

Interesante modificación de la GNU GPL que considera el caso de los programas que ofrecen servicios vía Web, o en general, vía redes de ordenadores. Este tipo de programas plantean un problema desde el punto de vista de las licencias robustas. Como el uso del programa no implica haberlo recibido mediante una redistribución, aunque el programa esté licenciado, por ejemplo, bajo la GNU GPL, alguien puede modificarlo y ofrecer un servicio en la red, usándolo sin redistribuirlo y por tanto, sin estar obligado a distribuir el código fuente.

La Affero GPL tiene una cláusula que obliga a que, si el programa tiene un medio para proporcionar su código fuente vía Web, no se pueda desactivar esa característica. Esto significa que si el autor original incluye esa capacidad en la fuente, cualquier usuario puede obtenerlo, y además esa redistribución está sometida a las condiciones de la licencia. **[idem, 5]**

➤ **IBM Public License Version 1.0**

Es una licencia que permite la redistribución binaria de trabajos derivados sólo si (entre otras condiciones) se prevé algún mecanismo para que quien reciba el programa, pueda recibir su código fuente. La redistribución de este se ha de hacer bajo la misma licencia. Esta obliga al que redistribuye el programa con modificaciones, a licenciar automáticamente y gratuitamente las patentes que puedan afectar a esas modificaciones, y que sean propiedad del redistribuidor. **[idem, 5]**

➤ **Mozilla Public License 1.1**

Esta posibilita que un programa o partes de él, brinden la opción entre ella y otras licencias. Lo que precisa es que si una parte de un programa consiente la GPL o cualquier otra licencia compatible con ella como alternativa, esa parte del programa se convierte en compatible con la GPL. **[7]** Esta es un ejemplo de licencia libre con origen en una empresa. Es una evolución de la primera licencia libre que tuvo el Netscape Navigator, y en su momento fue muy importante por ser la primera vez que una empresa muy conocida decidió distribuir un programa bajo su propia licencia libre. **[idem, 5]**

2. Licencias permisivas

Las licencias permisivas, a veces también llamadas liberales o minimalistas, no imponen prácticamente ninguna condición sobre quien recibe el software, y sin embargo ofrecen permiso de uso, redistribución y modificación. Este enfoque, puede entenderse como la garantía de las máximas libertades para quien recibe un programa. Entre estas licencias, una de las más conocidas es la licencia BSD, hasta el punto que en muchas ocasiones se refieren a las licencias permisivas como licencias tipo BSD.

➤ **Licencia BSD original**

La licencia BSD (Berkeley Database License) es considerada menos restrictiva que la GPL porque no incluye la cláusula **CL** (copyleft), dando así la posibilidad de crear versiones modificadas o derivadas de SWL que puedan convertirse con relativa facilidad en productos no libres o propietarios. Su fundamento parte de que si el SW es libre no debe imponer restricción alguna en su distribución, aunque ello implique que alguien lo use para su propio beneficio y no comparta el código fuente. **[idem, 5]**

➤ **Licencia de XWindows versión 11 (X11)**

Es la licencia usada para la distribución del sistema XWindows, el sistema de ventanas más ampliamente usado en el mundo Unix, y también en entornos GNU/Linux. Es una licencia muy similar a la licencia BSD, que permite redistribución, uso y modificación prácticamente sin restricciones. A veces, esta licencia es llamada “licencia MIT/X” (con poca precisión, porque el MIT ha usado otros tipos de licencias). Bajo esta licencia se distribuyen también trabajos derivados de X Windows, como Xfree86. **[idem, 5]**

➤ **Zope Public License 2.0**

Esta licencia (habitualmente llamada “ZPL”) es usada para la distribución de Zope (un servidor de aplicaciones) y otros productos relacionados. Es una licencia similar a la BSD, con el interesante detalle de prohibir expresamente el uso de marcas registradas por Zope Corporation. **[idem, 5]**

➤ **Licencia de Apache**

Licencia bajo la cual se distribuyen la mayor parte de los programas producidos por el proyecto Apache. Es similar a la licencia BSD. No tiene compatibilidad alguna con la GPL. Permite trabajar libremente con el código fuente, puede incluir el uso de productos propietarios. Sus cláusulas estipulan que productos derivados de un código Apache original, no pueden llevar como nombre Apache. Esta cláusula fue creada para evitar que cualquier versión modificada por

otros, de las distribuciones de esa organización se pudiera ofrecer a los interesados como una versión original oficial. **[idem, 6]**

Existen otras licencias como:

➤ **Licencias de documentación libre o FDL**

Si importante es liberar el SW también lo es liberar la documentación que le acompaña, trátase de manuales, documentación del código y todo lo que resulte necesario e importante para usar/modificar el programa o sus derivaciones. Es por estas razones elementales que aparecen las licencias para la documentación, para asegurar las libertades que proclaman los movimientos más avanzados del SWL.

La GNU FDL (GNU Free Documentation License) fundamentalmente permite que la obra pueda ser copiada, modificada y redistribuida. Esta licencia no hace especificaciones sobre el uso comercial y es del tipo CL. Admite precisar secciones invariantes dentro del texto, aunque esto puede ocasionar serios problemas al obligar así a que se conserven partes que pudieran resultar inconvenientes para ciertos usos. **[idem, 5]**

Muchas de las licencias de software libre están basadas en el copyright, en las que existen límites sobre las condiciones que se pueden imponer. Si una licencia basada en copyright respeta la libertad de acuerdo a lo antes mencionado, es poco probable tener otro tipo de problema que no se haya anticipado (a pesar de que esto ocurre ocasionalmente). Además, algunas están sustentadas por contratos y estos a su vez pueden imponer un rango mucho más grande de restricciones posibles. Lo que significa que existen muchas formas de que tal licencia pueda ser inaceptablemente restrictiva y que no sea libre.

En el caso de la incompatibilidad¹⁰ con las licencias libres pueden llevar con facilidad a proyectos inviables desde el punto de vista jurídico, así como en dependencia del modelo o estrategia económica y comercial

¹⁰ Una licencia es incompatible con la GPL cuando restringe alguno de los derechos que la GPL garantiza, ya sea explícitamente contradiciendo alguna cláusula, ya implícitamente, imponiendo alguna nueva.

al que este debe corresponder. Existen algunas licencias consideradas libres que de cierta forma presentan incompatibilidad con la GPL. **Ver Anexo 4**

A continuación se presenta una tabla que muestra los aspectos jurídicos que se deben tener presente a la hora de comercializar el SWL a nivel nacional e internacional. **[8]**

Nacional	Internacional
Ley 14 de 1977 sobre Derecho de Autor Artículo 35	Convenio de Berna para la Protección de las Obras Literarias y Artísticas, de 1979
Resolución Conjunta 1 de 1999 MINCULT y SIME. Artículo 11	Aspectos de la Propiedad Intelectual relacionados con el Comercio (ADPIC), de 1994

Tabla 1: Marco jurídico

1.3.3.2 Explotación de los derechos de autor y la propiedad intelectual del SWL.

La propiedad intelectual y en específico los derechos de autor son muy importantes en la industria del software; aunque éstos crean un ámbito legal, de gran reconocimiento de derechos exclusivos, no brindan la forma de explotar los mismos.

“La Propiedad Intelectual se define de forma amplia como los derechos que surgen del resultado de la actividad en el ámbito industrial, científico, literario y artístico”. **[9]**

Las facultades o derechos que se les reconocen a los autores se han dividido en dos grupos, los derechos morales o personales y los derechos patrimoniales. Los primeros están llamados a proteger los intereses intelectuales del creador con respecto a su obra. Los segundos están relacionados con la explotación comercial de la obra y la consiguiente obtención de beneficios económicos. Los derechos patrimoniales son: **[10]**

- Derecho de reproducción. Realizar copias de la obra.

- Derecho de distribución. Poner a disposición de terceros la obra, ya sea con fines gratuitos u onerosos, por ejemplo: alquiler, préstamo, licenciamiento.
- Derecho de comunicación pública. Dar a conocer la obra públicamente por medios que no consisten en la distribución de ejemplares.
- Derecho de transformación. Es una forma de explotar la obra por la cual el autor u otra persona previamente autorizada por el creador modifican la obra originaria haciendo surgir una obra derivada, ejemplo: traducciones, adaptaciones y actualizaciones, entre otras.

Características del Derecho de Autor:

- Es limitado en el tiempo.
- Concede un conjunto de facultades o derechos.
- Es exclusivo.
- Recae sobre obras intelectuales originales.



Figura 1: Creaciones de la Propiedad Intelectual [53]

1.4 Diferencias entre SWL y Software propietario.

Atendiendo a la definición de SWL se destaca que la contraposición de software libre es el software propietario. La mayoría del software propietario es comercial, pero el software libre puede serlo también, lo

que no contradice el espíritu del movimiento. Un ejemplo de software libre comercial es el usado por Red Hat¹¹, un conocido sistema operativo que se comercializa y es a su vez libre.

Muchos miembros del movimiento del software libre mantienen una posición extremista encaminada a eliminar totalmente el software propietario, comenzando por el Windows. Esta posición no se considera positiva, en cambio se debe tener en cuenta la necesidad de trazar una estrategia, tanto institucional como personal, para una migración al software libre, siempre que existan versiones equivalentes.

De acuerdo al artículo de Ted Schadler [11], la tabla 2 muestra las principales diferencias entre el software de código libre y el propietario.

ASPECTOS A CONSIDERAR	SOFTWARE PROPIETARIO	SOFTWARE LIBRE
Acceso al código fuente	<i>Prohibido por licencia</i>	<i>Sí - Garantizado</i>
Corrección de errores por el cliente-usuario	<i>No</i>	<i>Sí</i>
Duplicación del software	<i>Prohibido</i>	<i>Posible y Recomendado</i>
Libertad de competencia para el mantenimiento	<i>No - Depende del fabricante</i>	<i>Sí - Imposible limitarla</i>
Posibilidad de examinar el código del producto	<i>Prohibido</i>	<i>Sí</i>
Venta de segunda mano	<i>Prohibido</i>	<i>N/A</i>
Respeto a estándares globales	<i>En función del fabricante</i>	<i>En la mayoría de los casos</i>
Adaptaciones al cliente (p.ej. idioma)	<i>En función del fabricante</i>	<i>Disponibles</i>
Virus, gusanos	<i>Frecuentes</i>	<i>Muy poco frecuentes</i>

Tabla 2: Diferencias entre el software de código libre y el propietario

Estas diferencias tienen una gran importancia en la enseñanza universitaria porque se supone que las universidades tengan las potencialidades para enseñar tanto el software libre como el propietario. Este último favorece la dependencia tecnológica, no sólo de los futuros profesionales sino también de sus

¹¹ RedHat es una de las distribuciones de Linux más populares.

futuros clientes (o alumnos, en el caso en que dicho profesional incursione en la docencia), donde el uso de productos propietarios como “cajas negras” es totalmente incompatible con el concepto de “educación”, cambiándolo por el de “entrenamiento”.

Aunque la definición de “Software Libre” sólo data de unas décadas atrás, el concepto de libertad ha estado asociado desde siempre al desarrollo de las ciencias. Es así como han podido evolucionar la matemática, la física, y otras disciplinas. El Software Libre, al permitir el uso absoluto, el análisis y la reutilización de los programas impulsa fuertemente el intercambio de conocimientos y la colaboración entre programadores de todo el mundo.

1.4.1 Software propietario, privativo o cerrado

Los usuarios finales del software propietario no pueden tener acceso al código fuente del mismo, lo que hace imposible adaptar el producto a favor de sus necesidades, convirtiéndolos en clientes pasivos y dependientes de las empresas que los produce, comercializa y brinda ciertos servicios técnicos, perdiendo así sus principales derechos como consumidores.

Este tipo de software ha sido creado para mantener un Modelo de Negocio en busca de obtener mayores beneficios en lo que respecta el código fuente y el permiso legal para hacer cambios en los programas. Además se caracteriza por dividir a los usuarios y les dificulta colaborar con su trabajo y conocimientos en la mejora del software y en pro del bien común. **[idem, 12]**

1.4.2 Software de Código Abierto (SCA)

Open Source o SCA

“Open Source es una forma para conseguir que la gente colabore con el SW sin verse envueltos en problemas de propiedad intelectual, evitando tener que negociar contratos cada vez que compran SW, con muchos abogados de por medio. En general, solo se quiere que el SW funcione y ser capaces de tener

personas que contribuyan con soluciones, entre otros. Así que se sacrifican algunos derechos de propiedad intelectual, y simplemente se permite que todo el mundo use ese SW.” [13]

Se define el concepto de Código Abierto (Open Source) como sigue: [14]

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 solo 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.

1.4.2.1 Polémica entre SWL y Fuente Abierta o Código Abierto

La Fundación de Software Libre (FSF) con su licencia GPL y OSI¹² con Código Abierto defienden al SWL bajo cualquier motivo donde las ideas de la OSI pueden ser más avanzadas en algunos aspectos. Estos dos movimientos progresistas se diferencian en sus Modelos de Negocio y los procesos de desarrollo desde el punto de vista de la Ingeniería del Software.

¹² OSI: Open Source Initiative

El movimiento del SWL hace especial énfasis en los aspectos morales o éticos del software, viendo la excelencia técnica como un producto secundario de su estándar ético. El movimiento de SCA ve la excelencia técnica como el objetivo prioritario, siendo la compartición del código fuente un medio para dicho fin. Por dicho motivo, la FSF se distancia tanto del movimiento de código abierto como del término "Código Abierto" (en inglés Open Source), ya que la OSI sólo aprueba las licencias que se ajustan a la OSD (Definición de Código Abierto), la mayoría de la gente lo interpreta como un esquema de distribución, e intercambia libremente "código abierto" con "software libre". Aun cuando existen importantes diferencias filosóficas entre ambos términos, especialmente en términos de las motivaciones para el desarrollo y el uso de tal software, raramente suelen tener impacto en el proceso de colaboración.

Además, la FSF aporta grandes posibilidades para la libertad del software pero tiene un lento modelo de desarrollo que ya es necesario cambiar, mejorar y actualizar en alguna forma. No es menos cierto que la FSF ha ido adoptando nuevas versiones de la licencia GPL, justamente la GPL V3 en el año 2007, e incorporando cosas nuevas a la documentación, lo que constituye un reconocimiento de las necesidades que pone en práctica el movimiento Open Source representado por la OSI.

El éxito de este sistema es solo una ventaja práctica para quienes no ven el sistema en términos éticos y creen que su fin es su éxito, desafortunadamente muchas de las distribuciones de GNU/Linux contiene software propietario y eso es un problema ya que no contribuye a la razón por la cual se diseñó. En la gráfica 1, tomada del sitio de Najaraba.com se puede ver que el "free software" (azul) se busca mucho más, pero "open source" (rojo) genera muchas más noticias, ya que es el término más utilizado en la mayoría de productos empresariales, dentro de los cuales, unos son libres y otros no. Es el término más aceptado empresarialmente. **[15]**

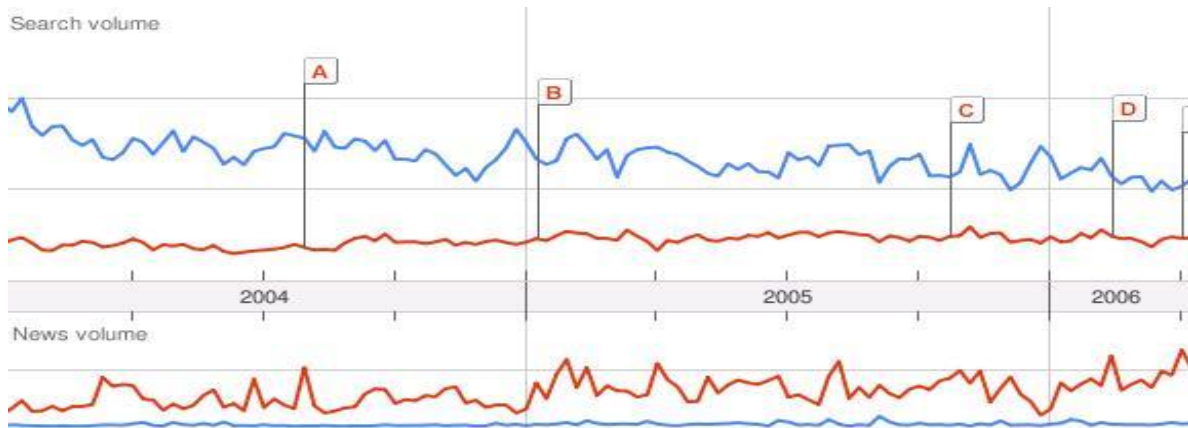


Tabla 3: Comparativa por niveles de búsqueda entre “free software” (azul) y “open source” (rojo)

1.5 Modelos de Negocio para la Comercialización de SWL. Tipos identificados en la búsqueda bibliográfica.

1.5.1 El Modelo de Negocio

Un Modelo de Negocio según su estrategia, se basa en la oportunidad de mercado, los productos y/o servicios que pueden ser ofrecidos, las dinámicas de competitividad y las opciones estratégicas para evolucionar. Los recursos para aplicarla se crean por medio del personal y los colaboradores, organización y cultura, operaciones, marketing y ventas, gerencia de procesos, entre otros; y de esta manera medir los beneficios para los inversionistas a través del desempeño financiero, marca y reputación, y cuota de mercado, con el fin de tener grandes perspectivas de crecimientos e indicadores económicos. En la siguiente figura 2 se observa lo antes explicado: **[16]**



Figura 2: Concepto, Valor y Capacidades del Modelo de Negocio.

En fin, un Modelo de Negocio es el mecanismo por el cual la empresa crea valor y lo transforma en ganancias, en general como un negocio trata de generar ingresos y ganancias. Resume la forma en que la compañía planea servir a sus clientes. Implica el desarrollo de la estrategia de negocios y su aplicación.

Responde a las siguientes cuestiones:

- ¿Cómo seleccionará sus clientes?
- ¿Cómo define y diferencia sus ofertas de producto?
- ¿Cómo crea utilidad para sus clientes?
- ¿Cómo consigue y conserva a los clientes?
- ¿Cómo sale al mercado (estrategia de publicidad y distribución)?
- ¿Cómo define las tareas que deben llevarse a cabo?
- ¿Cómo configura sus recursos?
- ¿Cómo consigue el beneficio? **[17]**

Es importante tener presente que a veces se confunde el término de Modelo **de** Negocio, que no es lo mismo que Modelo **del** Negocio, ya que el primero se refiere a la “forma de hacer negocios”, es decir, como lo explicado anteriormente y el segundo, al modelo de Ingeniería de Software que describe los

procesos de negocio, identificando quiénes participan y las actividades que requieren automatización, con sus respectivos actores y casos de usos.

¿Cómo hacer negocios exitosos con SWL?:

Para dar respuesta a esta pregunta Román H Gelbort en un Café Conferencia en el 2007, trató el tema Modelos de Negocio con SWL donde expuso una serie de elementos que sirven como guía a la hora de hacer un modelo exitoso:[18]

1. visualizar como obtener ganancias.
2. buscar las formas para que el negocio sea sustentable.
3. tratar de mantener un equilibrio desde el inicio.
4. comenzar desde puntos intermedios para ahorrar costos.
5. desarrollar trabajo modular, es decir el código debe estar bien modularizado.
6. reconocer códigos de buena calidad.
7. aprender la comunicación con la interfaz humana.
8. desarrollar la solución en conjunto con el cliente.
9. cumplir con las tareas de la empresa.
10. lograr una capacidad de adaptación a las necesidades del cliente
11. dar una estructura jerárquica a la empresa en la que todos ganen.
12. aprender a comunicarse con el cliente.
13. tener en cuenta que las maneras de difusión del SWL deben ser innovadoras y motivadoras.
14. lograr una venta transaccional del producto terminado.
15. organizar bien la documentación del producto.
16. mantener el SW actualizado.
17. dar mantenimiento y soporte.

1.5.2 Diferencias entre los Modelos de Negocio para SW propietario y SWL

Actualmente, las tecnologías de la información se han convertido en la base fundamental para el negocio de SW, proporcionando valor añadido y una inversión esencial. El tema de discusión de muchos desarrolladores es cómo sacar el máximo rendimiento a esas tecnologías. Sin embargo, la aparición del software libre como una alternativa real ha activado este debate.

La seguridad en el código abierto, está en la amplitud de uso del propio programa. En el modelo comercial se deposita la confianza en una empresa, de la que se conoce su credibilidad y respuesta ante las necesidades de los clientes.

Entre los dos Modelos de Negocio, la rivalidad que existe está influenciada por el sentimiento anti-Microsoft que existe en numerosas comunidades de código abierto. Esto ha ocasionado una inquietud en determinadas personas que ven el software libre como la vía de terminar con el dominio de Microsoft.

El Modelo de Negocio para Software Libre se fundamenta principalmente por los servicios que ofrecen los desarrolladores. Existen cientos de distribuciones Linux en todo el mundo, lo que hacen es empaquetar diferentes aplicaciones creando un sistema operativo libre, puesto al mercado por diferentes canales de distribución. El desarrollo de Software Libre y de modelos de distribución puede aumentar la visibilidad del desarrollo del software local y de sus programadores. **[idem, 17]** Ver figura 3 y 4.



Figura 3: Modelo de Negocio del software propietario

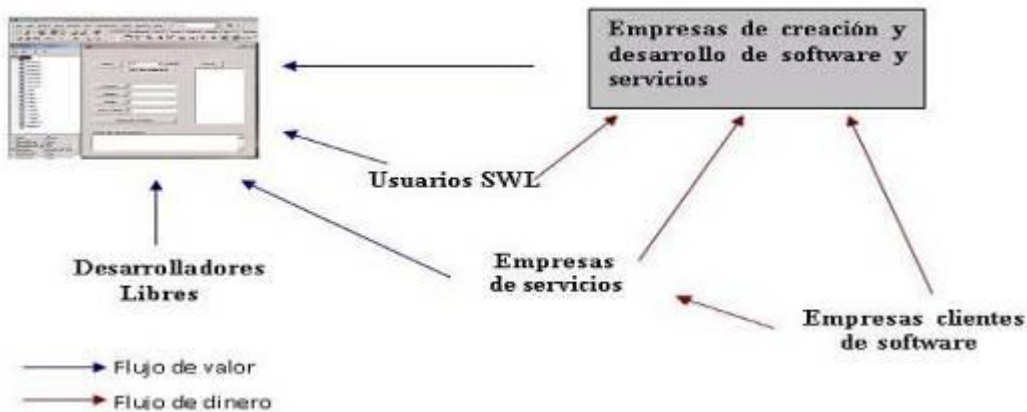


Figura 4: Modelo de negocio de software libre

El modelo del software propietario se basa en que una determinada empresa, desarrolla el software de código cerrado por el cual adquiere beneficios económicos mediante la venta del propio producto y las nuevas versiones, que pueden ser vendidas a un usuario y a las empresas clientes; también, a través de un contrato con la empresa dedicada a los servicios, la empresa desarrolladora, le paga una cierta cantidad de dinero a esta empresa por publicar los servicios que están asociados al producto, para que luego se los ofrezca de forma personalizada a sus empresas clientes. Por otro lado está el modelo de software libre donde la empresa que diseña el producto, permite que este sea copiado, modificado y estudiado por desarrolladores libres, usuarios y empresas dedicadas a servicios, de acuerdo a sus necesidades. El autor del software puede, en algunos casos, imponer ciertas condiciones bajo algunas licencias en cuanto a modificar el código, puede obtener ingresos a través de varios servicios (como soporte, adiestramiento, consultoría, entre otros), que son comercializados con los usuarios, empresas clientes y empresas dedicadas a los servicios, éstas últimas, en algunas ocasiones adquieren el SWL para modificarlo y brindar servicios, asociados a él, a las empresas clientes mediante el pago por suscripción u otra forma.

A raíz de todo esto surge la idea de que el SCA se convierte en software como servicio [19]. Pueden existir empresas que desarrollen el SWL y ofrezcan servicios de consultoría, formación, y soporte.

1.5.3 Diferentes Modelos de Negocio para SWL

Cuando se utiliza Software Libre para desarrollar una aplicación, se puede modificar, adaptar o corregir el software y cualquier empresa puede dar soporte sobre un determinado producto, ya que el código es visible para todos, creándose una competencia entre los proveedores de servicios alrededor del SWL. Esto presenta la base para la construcción de Modelos de Negocio más adaptados a las necesidades del mercado, necesariamente relacionados con el desarrollo de aplicaciones a la medida, donde los productos de SWL pueden segmentarse en función del tamaño de las entidades a las que se orienta su público objetivo. A partir de este tamaño y de las características específicas del Software Libre se puede describir como se desarrolla la competencia en este mercado.

En esta sección se describirán diferentes Modelos de Negocio. Muchos se basan en SWL puro, pero los relacionados con la combinación de Software Libre y software propietario son más exitosos, donde los más relevantes se logran cuando los dos actores principales detrás del desarrollo de una determinada tecnología (el propietario del software y la comunidad) se encuentran motivados para mantener y hacer evolucionar dicha tecnología.

Luego de una búsqueda en fuentes como Internet, revistas electrónicas, libros y archivos relacionados con el tema de los Modelos de Negocio para comercializar SWL, que actualmente existen en el mundo, se presenta un resumen y análisis de los modelos que más se relacionan con las fichas descritas en el **Anexo 1**, a través de varios aspectos que se tuvieron en cuenta a la hora de estudiarlas.

Modelo 1: Comercialización con licencias duales.

Relación: Este modelo está estrechamente relacionado con las fichas 1.1 y 1.2.

Aplicado en: Open Office, Trolltech, MacOS X

Descripción: Una licencia dual es un Modelo de Negocio en la que una compañía que comercializa un determinado producto aporta a sus licencias dos modelos a escoger: código abierto y código “cerrado” o propietario. Aquí, el cliente puede elegir el tipo de licencia con las condiciones de una licencia de código abierto, como la de tipo BSD o la GPL, que esta última permite crear múltiples niveles de sublicencias en forma de código abierto (normalmente bajo idénticas condiciones) y otra con licencia de código cerrado.

Ventajas: La ventaja de pagar una licencia por un producto que también se pudiera adquirir con licencia gratuita, es que se puede utilizar el producto para integrarlo dentro de un software propietario y luego distribuirlo con licencia propietaria. Por ejemplo, con las licencias tipo BSD es posible re-licenciar el código derivado a cualquier tipo de licencia, incluidas licencias propietarias.

Desventajas: Para los clientes, la licencia propietaria establece ciertas condiciones restrictivas en caso de que desee redistribuir o modificar el producto.

Comentario: Este modelo se ha comprobado que es exitoso porque las licencias duales estimulan la introducción del software de código abierto en los negocios mientras impulsa su desarrollo con los ingresos que brinda la venta de licencias propietarias.

Modelo 2: Comercialización de servicios asociados a un producto.

Relación: Este modelo está estrechamente relacionado con las fichas 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.7 y 2.8.

Aplicado en: Alcove, Open Source Risk Management, Red Hat, Askimet, QINDEL GROUP en España, Yaco

Descripción: Este Modelo de Negocio consiste en desarrollar una determinada aplicación bajo licencia libre, ofreciendo servicios específicos sobre ese software. Entre los servicios que se pueden ofrecer están:

1. El soporte, consiste en mantener el software instalado y actualizado.
2. La formación y consultoría puede estar ofertada con la implantación del software, cuando se actualiza y se introducen nuevas funcionalidades o simplemente para aumentar habilidades de los usuarios.
3. La personalización es proporcionada por las compañías expertas en el proceso de compilar e integrar productos de código abierto implantados en el sistema de destino.
4. Los entrenamientos pueden ofrecerse con la introducción del software, cuando este es actualizado a través de adiestramiento para expandir las habilidades.
5. Con el control de riesgo y seguridad se pueden identificar y moderar los riesgos que el código pueda tener, para brindarle una mayor seguridad y calidad al software libre a la hora de su comercialización.

Ventajas: Aporta un gran valor añadido a las empresas y la solución de código abierto es más flexible para estos tipos de servicios que para el uso de software propietario. Puede ser llevado a cabo dentro de la propia compañía o a través de un proveedor de servicios externo.

Desventajas: Los ingresos son a mediano plazo, por las previas inversiones que se deben asumir.

Comentario: A pesar de que se debe invertir al principio y de que los ingresos no son inmediatos, tiene grandes beneficios principalmente de los que pagan por un servicio más profesional. Suelen aparecer empresas que se especializan en un dominio concreto ofreciendo servicios de consultoría, desarrollo y personalización, formación, entre otros, sobre el mismo.

Modelo 3: Micro-pagos por suscripciones.

Relación: Este modelo está estrechamente relacionado con las fichas 3.1.

Aplicado en: Alfresco, Linux Weekly News (revista)

Descripción: Este Modelo de Negocio consiste en que el cliente que utiliza el software libre paga por un conjunto de servicios durante un tiempo determinado, a modo de suscripción. Algunos negocios (sobre todo las publicaciones digitales) basan su estructura de ingresos en las suscripciones de sus usuarios.

Ventajas: Puede ser modificado en función de las necesidades propias de cada proyecto. El pago de suscripciones por parte de los socios tiene ventajas sobre los usuarios normales.

Desventajas: Este Modelo de Negocio está lejos de ser maduro y necesita de un gran número de usuarios para que funcione correctamente.

Comentario: Con este modelo muchas empresas pueden hacer sus negocios mediante las suscripciones, pagando una tasa anual, donde se le permite tener acceso a información privilegiada, certificación, entre otros.

Modelo 4: Pagos por donaciones.

Relación: Este modelo está estrechamente relacionado con la ficha 4.1 y 4.2

Aplicado en: eMule, Gimp, Wikipedia Foundation, Firefox, Apache

Descripción: Se basa en aportes voluntarios de grandes empresas y de pequeños aportes de una gran cantidad de usuarios, estos aportes son su principal fuente de ingresos.

Ventajas: Garantiza una regularidad en el flujo de dinero y permite tener un horizonte más despajado ya que la mayoría de los acuerdos se concretan por un cierto número de años.

Desventajas: Este Modelo de Negocio está lejos de ser maduro y necesita de grandes aportes para que funcione correctamente. Existe la posibilidad de que un proyecto termine por depender de la buena voluntad de una compañía, que puede cancelar la financiación cuando le parezca mejor. Es más

inestable y sostener el desarrollo de un emprendimiento de cierta envergadura sobre esta base puede ser demasiado arriesgado.

Comentario: Tienen que aportar productos de alta calidad para que las empresas importantes no manchen su imagen, es decir no comprometan su nombre.

Modelo 5: Desarrollo de componentes comerciales para productos de software libre.

Relación: Este modelo está estrechamente relacionado con las fichas 5.1 y 5.2.

Aplicado en: Zope Corporation, Ximian, OpenSystems

Descripción: Se basa en el desarrollo bajo licencia libre de aplicaciones específicas, con un objetivo específico, en las que se desarrollan determinados componentes comerciales que cubren determinadas necesidades de un cliente. Consiste en repartir paquetes compuestos de un conjunto de componentes probados e integrados.

Ventajas: Es posible construir aplicaciones acorde a las necesidades del cliente mediante la integración de los elementos. El código puede ser rápidamente modificado para adaptarse a nuevas plataformas. Es mucho más flexible.

Desventajas: Para desarrollar estos componentes deben tener sus aplicaciones fundamentalmente sobre plataformas libres.

Comentario: Utilizando componentes basados en software libre se puede partir desde un código existente y no desde cero. Es posible beneficiarse del proceso de revisión y actualización de versiones de cada proyecto individual basado en software libre sin realizar esfuerzos importantes en reintegrar las modificaciones.

Modelo 6: Cobrar por el SWL.

Relación: Este modelo está estrechamente relacionado con las fichas 6.1, 6.2 y 6.3.

Aplicado en: Ada Core Technologies, Aladdin Enterprises, Suse

Descripción: Se puede aplicar de diversas formas:

Para productos que se distribuyan bajo licencias libres: el software es libre desde que el productor lo comienza a distribuir durante un tiempo sólo a sus clientes, que le pagan por ello (Por ejemplo proporcionando el código fuente a sus clientes junto con los binarios, y cobrándoles por acceder al paquete formado por el código fuente, los binarios y los servicios profesionales. También, vendiendo el

software en un CD junto a un manual de usuario.), y luego lo publica de manera gratuita.

Para productos que se distribuyan bajo licencia propietaria por un tiempo: el software se distribuye bajo una licencia propietaria durante un período y luego libre donde pasado un tiempo, esa versión pasa a distribuirse con una licencia libre. (Por ejemplo se puede crear una licencia que permita la libre distribución, prohíba el uso comercial y garantice que el producto adquiriera la licencia GPL después de un año o si el vendedor quiebra).

Ventajas:

Para productos desarrollados bajo licencias libres: aplicable a todas las licencias de código abierto y a la que se ajustan muchas compañías. La empresa productora obtiene ingresos de los clientes interesados en disponer lo antes posible de nuevas versiones, y a la vez minimiza la competencia, ya que cualquier empresa que quiera competir usando ese producto, sólo podrá hacerlo con la versión libre.

Para productos que se distribuyan bajo licencia propietaria por un tiempo: los clientes se aseguran de que el producto es adaptable a sus necesidades ya que disponen del código. El productor obtiene ingresos de sus clientes, que perciben esta disposición preferente del software como un valor añadido.

Desventajas: Debido al retraso con el que el producto está disponible para la comunidad del software libre, es prácticamente imposible que ésta pueda colaborar en su desarrollo, por lo que el productor se beneficia muy poco de contribuciones externas.

Comentario: El modelo sólo funciona si los clientes no hacen público el programa cuando lo reciben. En estos casos las empresas desarrolladoras obtienen los beneficios mencionados, pero no a coste cero.

Modelo 7: Venta de SWL por imagen de marcas.

Relación: Este modelo está estrechamente relacionado con las fichas 7.1 y 7.2.

Aplicado en: ERP Openbravo, Red Hat.

Descripción: Es utilizado por empresas que invierten en establecer una marca con buena imagen, y bien reconocida, que más adelante les permita vender con suficiente margen productos libres. Estas empresas tratan de vender algo que en general se puede obtener a un coste bastante menor en la Red (o en otras fuentes con menos imagen de marca). Por ello, han de conseguir que el consumidor reconozca su marca, y esté dispuesto a pagar el sobreprecio.

Ventajas: Pueden ofrecer una gran cantidad de servicios, tratando de rentabilizar al máximo esa imagen de marca.

Desventajas: Las empresas tienen que invertir en publicidad.

Comentario: En muchos casos no sólo venden esos productos, sino que los acompañan de servicios que los clientes acepten como valor añadido.

Modelo 8: Hardware con SWL incluido o embebido.

Aplicado en: Neoteris, Maemo

Relación: Este modelo está estrechamente relacionado con la ficha 8.1.

Descripción: Se basa en software libre incluido en Hardware, mediante la adopción de estándares, de un Kernel estable y de nuevas tecnologías soportadas.

Ventajas: La combinación de SWL con un hardware específico es también una buena oportunidad para los editores de SWL para crear nuevos productos.

Desventajas: En ocasiones muchas empresas que se dedican a esto necesitan asociarse a otra que se dedique a esta parte del hardware, ya que ellas no están adentradas en este campo y instalar el SWL en el para luego venderlo. Otra es que los fabricantes de hardware necesitan capacitación en SWL para incluir el SO y componentes a este.

Comentario: Por ejemplo, se puede adaptar el núcleo Linux a las exigencias de lo embebido (poco consumo de memoria, y su portabilidad sobre diferentes tipos de microprocesadores).

Modelo 9: Software propietario sobre Software libre.

Relación: Este modelo está estrechamente relacionado con la ficha 9.1.

Aplicado en: Google, Oracle

Descripción: Consiste en construir software propietario utilizando herramientas libres, y después vender sus servicios, a través de Internet.

Ventajas: Estos se han desarrollado gracias a las plataformas libres que hacen posible elaborar productos sin invertir en costosas plataformas propietarias, ni tener que realizar complicados desarrollos desde su inicio.

Desventajas: No logra captar las ventajas competitivas derivadas de Modelos de Negocio puramente basados en software libre por lo que pueden llegar a tener dificultades para obtener rentabilidad a largo

plazo.

Comentario: Con este modelo muchas empresas intentan la integración con el software libre, utilizando piezas de código cerradas para tratar de mejorar la compatibilidad y permitir la ejecución de su software sobre sistemas operativos libres.

➤ **Empresas comercializan sus productos basados en SWL**

Muchas empresas comercializan sus productos basados en SWL a través de los servicios, por ejemplo:

1. Empresas que distribuyen el software libre.
2. Empresas que dan soporte, consultoría o formación al software libre.
3. Empresas que se basan en programas de código abierto.
4. Empresas que utilizan programas de código abierto como usuarios finales.
5. Empresas que venden accesorios para el software libre.

Más adelante, se muestra la clasificación que Guillermo Armelini y Francesc Miralles le dan a estos tipos de empresas, que es bastante parecida a la ejemplificada anteriormente.

1.5.4 Otras clasificaciones de Modelos de Negocio sobre SWL.

1. Clasificación de Miralles y Armelini

Según Francesc Miralles y Guillermo Armelini, las empresas utilizan diferentes Modelos de Negocio **[20]**:

1) Empresas que distribuyen el software libre.

Aunque el software libre puede ser conseguido de diversas maneras, una empresa puede crear su propia versión y ofrecer servicios añadidos. El ejemplo más claro son las distribuciones de Linux. Este sistema operativo requiere una compleja instalación, y numerosas descargas desde Internet de sus partes. Existen

empresas que han creado su propia versión facilitando la instalación y administración de Linux, ofreciendo así prestaciones adicionales con las que hacen negocio.

2) Empresas que den soporte, consultoría o formación en software libre.

El software libre también necesita de profesional calificado y de un soporte adecuado que responda ante las necesidades de los clientes. Las empresas distribuidoras son las que dominan el mercado de la consultoría y asesoramiento o formación.

El Modelo de Negocio de **Red Hat** ofrece a sus clientes software empresarial en el cual esta añadido el costo de soporte para el servidor donde se instale, si es requerido instalar el software en varios servidores debe abonarse una cantidad igual por cada uno de ellos. Se sustenta en la prestación de un servicio de asistencia técnica profesionalizando a empresas de todo tipo.

3) Empresas que se basen en programas de SCA.

Muchas empresas renuevan sus productos para que funcionen sobre plataformas de código abierto con el fin de independizarse de las empresas proveedoras. Pueden aumentar sus márgenes, y competir en aquel mercado dónde sus posiciones son más ventajosas.

SAP y Oracle han adaptado sus productos para Linux.

4) Empresas que utilizan programas SCA como usuarios finales.

Pueden aportar modificaciones a la comunidad. Estas empresas no generan ingresos por el software libre, pero lo utilizan porque les aporta un mayor valor añadido, o bien por que aumenta sus márgenes operativos. También podrían aportar un valor añadido al software libre, realizando las modificaciones necesarias para ellas en el software y posteriormente compartiéndolas con la comunidad.

5) Empresas de venta de accesorios para el software libre.

Las empresas en el mundo del software de código abierto pueden generar ingresos por varios métodos, como es la venta de accesorios:

- Ventas de productos relacionados: libros, documentación, objetos de marca, CD, entre otros.

eMule es una empresa que posee una tienda donde vende todo tipo de accesorios como pullovers, CD, entre otras cosas.

2. Clasificación de Hecker

Esta clasificación fue la más usada por la publicidad de la Open Source Initiative. Incluye varios modelos poco centrados en software libre (en ellos es poco más que un acompañante del modelo principal). En cualquier caso, los modelos que describe son los siguientes [21]:

Venta de servicios relacionados con el producto (Support seller). La empresa promueve un producto libre (que ha desarrollado o en cuyo desarrollo participa activamente) y vende servicios, como consultoría o adaptación a necesidades concretas para él.

Venta de otros productos propietarios (Loss leader). En este caso, el programa libre se utiliza para promover de alguna forma la venta de otros productos propietarios relacionados con él.

Venta de hardware (Widget frosting). El negocio fundamental es la venta de hardware, y el software libre se considera un complemento para él, que puede ayudar a la empresa a obtener una ventaja competitiva.

Venta de accesorios (Accessorizing). Se comercializan productos relacionados con el software libre, como libros, dispositivos informáticos, entre otros.

Venta de servicios (Service enabler). El software libre sirve para crear un servicio (normalmente accesible en línea) del que la empresa obtiene algún beneficio.

Venta de derechos por uso de marcas (Brand licensing). Una empresa registra marcas que consigue asociar con programas libres, probablemente desarrollados por ella. Luego obtiene ingresos cuando vende derechos del uso de esas marcas.

Vende, libera (Sell it, free it). Modelo similar a loss leader, pero realizado de forma cíclica. Primero se comercializa un producto como software libre. Si se consigue que tenga cierto éxito, la siguiente versión se distribuye como software propietario durante un tiempo, y pasado éste también se libera. Para entonces, se empieza a distribuir una nueva versión propietaria, y así sucesivamente.

Franquicia de software (Software franchising). Una empresa socia otorga licencias o marcas, relacionadas con un programa libre determinado.

3. Clasificación de Feller

Después de consultar “Perspectives on Free and Open Source Software” se muestra a continuación la clasificación que Feller le da a los Modelos de Negocio: **[idem, 2]**

El Distribuidor: proporciona el acceso al código fuente y al software. Estos obtienen ganancias distribuyendo el producto a través de un CD o en un lugar como una descarga online, y ofreciendo soporte a los servicios para las empresas clientes, para asegurar que ellos obtengan la última actualización.

El Productor del Software (GPL Model): el productor del software se esfuerza en desarrollar el código abierto para que el producto esté disponible para el usuario final. Este modelo está dividido en los productores de software que usan licencias libres, y los productores de software que no usan licencias libres.

El Proveedor de Servicios Tercerizados: La misión de los proveedores de servicios tercerizados es simple. Ellos no cuidan si el usuario consiguió el código o el producto de otra empresa o de un tercero. Si el producto que está usando cumple con una serie de requisitos, ellos le darán el soporte que necesita.

4. Clasificación según IT Managers Journal Ver el Anexo 2.

- 1. La Estrategia de Optimización:** Se refiere a intentar mantener el SWL en condiciones óptimas ya sea a través de servicios de soporte y mantenimiento.
- 2. La Estrategia de Licencia Dual:** Se refiere al desarrollo de SWL bajo dos tipos de Licencia una propietaria y otra libre.

3. **La Estrategia de Consultoría:** Se refiere a los servicios de consultoría que brindan las empresas por la personalización (customización) de un software o del desarrollo de un software a la medida.
4. **La Estrategia de Suscripción:** Se refiere a pagar por un servicio durante un tiempo determinado a modo de suscripción.
5. **La Estrategia del Patrocinio:** se refiere al aporte voluntario de grandes corporaciones a proyectos de SWL gestionados por fundaciones.
6. **La Estrategia de Almacenamiento:** Se refiere a almacenar productos de SWL en un repositorio.
7. **La Estrategia Embebida:** Se refiere al SWL incluido en Hardware.

Después de analizar los Modelos de Negocio expuestos y las distintas formas de enfocarlos se observa cómo el SWL es una alternativa para muchas empresas, que aspiran competir con sus productos en el mercado y por consiguiente en erradicar la dependencia tecnológica a la que están sometidas. Cabe destacar de que son muchos, pero solo los más relacionados son los que aparecen en esta sección, además de las diferentes clasificaciones que existen como la de: Hecker, Armelini, Feller, entre otros, pudiendo destacarse que coinciden en general.

1.6 Importancia de comercializar SWL en el mercado.

El uso de software libre en el sector empresarial tiene una gran importancia porque supone un cambio de mentalidad sobre un modelo (modelo de venta de licencias) ya asumido. Además, permite a una empresa adaptar soluciones a sus necesidades, arreglar fallos operativos o de seguridad. Ayuda al cliente elegir el mejor proveedor de servicios, donde la disponibilidad del código fuente y la capacidad de modificarlo, permiten que una empresa no quede atada a ningún proveedor. También, se puede llegar a eliminar la existencia de puertas traseras, troyanos, entre otros, debido a que cada empresa puede auditar el código fuente de las aplicaciones. El uso sistemático de estándares, hacen muy difícil que una determinada aplicación pueda quedarse sin soporte. En caso de que esto ocurra, la disponibilidad del código fuente deja que otro grupo de usuarios o empresas pueda liderar el desarrollo de la misma.

Desde el punto de vista empresarial el desarrollo de Software Libre tiene una serie de ventajas como:

- La disponibilidad de una comunidad potencial enorme de programadores y testadores del software.
- Posibilidad de respuesta rápida ante clientes por posibles problemas en el código.
- Evolución automática de las piezas de software.
- Tendencia a una calidad enorme en el código desarrollado: consecuencia de la necesidad de trabajar en equipo con personas especializadas.
- Tendencia a la utilización de estándares, que permite la construcción de soluciones mucho más completas.

Es difícil para una empresa que invierte recursos en el desarrollo de software la decisión de liberar código. El éxito de la iniciativa depende en gran medida de la utilidad real que tenga ese software para la mayoría de los usuarios.

1.7 Aspectos económicos, académicos y éticos que se deben tener en cuenta.

Aspecto económico

Se sabe que la libertad del software no implica su gratuidad, esto no revoca el hecho de que el software propietario es altamente costoso tanto en su adquisición como en su mantenimiento.

Por eso, es necesario resaltar una diferencia entre ambos modelos: en el caso del Software Propietario, lo que se adquiere al pagar el precio de una licencia es el permiso de ejecutar el programa en cuestión bajo determinadas condiciones, en tanto que al adquirir un producto de Software Libre se obtiene una copia del programa (incluyendo su código fuente) y el permiso del autor para hacer uso de las cuatro libertades que definen a este tipo de software.

Aspecto académico

En el caso del software propietario, a través de la historia, una serie de empresas han querido marcar el rumbo tecnológico. No obstante, estas empresas han tratado de ejercer presión sobre el sistema

académico para lograr el acercamiento de los estudiantes a sus productos, invirtiendo en muchos casos elevadas sumas de dinero a tal efecto (bajo la forma de convenios o donaciones).

El uso de herramientas propietarias, ligadas a tecnologías específicas bajo el control de determinadas empresas, contribuye a lograr una fuerte dependencia del futuro profesional en las mismas. Después, este profesional en el desempeño de sus actividades, debe actuar como un agente de ventas de dicha empresa al ofrecer a sus clientes soluciones basadas exclusivamente en los productos que conoce.

Por otra parte, al utilizar herramientas libres, se le brinda al estudiante la posibilidad de conocer los detalles de la implementación de las mismas y los fundamentos tecnológicos en los que éstas se basan.

A modo de ejemplo, se puede nombrar el caso de materias como “Compiladores”, “Sistemas Operativos”, “Redes y telecomunicaciones”, “Máquinas Computadoras”, entre otras, las cuales no podrían establecerse correctamente sin la existencia de herramientas libres que permitan analizar las distintas implementaciones y utilizarlas para aplicar los conceptos teóricos. Es preciso notar que la mayoría de estos programas han surgido del ámbito académico.

Aspecto ético

Para entender al software libre hay que hablar de temas éticos y sociales. Las libertades no existen por ser: es una creación humana. La civilización construye nuevas libertades y la preocupación de la comunidad del software libre son las personas.

La Ética es una de las ramas más importantes de la filosofía. Descubrir racionalmente lo que es “bueno” para la sociedad y para los individuos de una forma equilibrada.

Numerosos seguidores del código abierto dan razones éticas para el desarrollo masivo del software libre, como Richard Stallman, que incluso promueve que todo el software debería ser libre [22]. Pero esto sobrepasa las barreras del conocimiento de una empresa, donde se fusionan el planteamiento estratégico y las bases culturales y éticas. Es una cuestión personal, la visión del software libre como modelo ético, que corresponde a cada persona decidir.

Con relación al alto costo de las licencias de uso, es una actividad muy común en la actualidad, la utilización de “copias no autorizadas” de productos propietarios (a lo que se le suele asignar el desafortunado nombre de “piratería”). La práctica de este tipo de actividades conlleva:

- El uso de copias no autorizadas contribuye a la estrategia utilizada por las empresas productoras de software propietario para promocionar sus productos y provocando así el llamado “efecto de red”, ya que al tratarse de tecnología cerrada y bajo el control de dichas empresas, se está propiciando el establecimiento de la misma como un estándar “de facto”.

1.8 Conclusiones

En el transcurso de este capítulo se mostraron las características del software libre desde el punto de vista empresarial. Como resumen se enumeran una serie de conclusiones:

1. El Software Libre es una alternativa viable para todas aquellas empresas que desean lograr su independencia tecnológica y ampliar sus horizontes, en aras de obtener posiciones ventajosas en el mercado.
2. Es recomendable utilizar licencias libres a la hora de distribuir el SWL, bajo el respeto de las cuatro libertades, así como saber la compatibilidad entre ellas para no caer en problemas éticos y legales.
3. El Modelo de Negocio de SWL puede ser aplicado a diferentes productos/servicios antes de comenzar su comercialización para escoger la vía más rentable para obtener sus beneficios.
4. Segmentar su mercado o identificar diferentes grupos dentro de su mercado, es una buena estrategia a la hora de utilizar diferentes tácticas de marketing cuando venda sus productos a diferentes clientes.
5. El análisis de los diferentes Modelos de Negocio encontrados permitió conformar una serie de fichas e integrarlas en dependencia a los servicios que más se relacionaban, logrando una correspondencia uniforme y bien estructurada de estos modelos.
6. La adopción del SWL es importante para buscar nuevas fuentes de financiación, nuevas puertas en el mercado, así como una mayor orientación a las necesidades del cliente.

CAPÍTULO 2: EL SWL EN AMÉRICA LATINA. LA UCI COMO CAMPO DE ACCIÓN.

2.1 Países de América Latina vinculados al SWL

El software libre está conquistando América Latina, en la sustitución de los programas informáticos propietarios por otros de código abierto. A diferencia de lo que ocurre en Europa, en América Latina muchos países están completamente seguros de que el único camino para impulsar la sociedad de la información como palanca para el desarrollo consiste en usar SWL. Como son Argentina, Brasil, Cuba, Chile, Venezuela, México, entre otros. En el caso de este último, los programas sin patentes se han incluido en un plan de austeridad que aplica el gobierno federal, debido al casi nulo costo que tienen en su aplicación. [23]

El ahorro de costos que supone no pagar licencias por el uso de programas propietarios, no es el motivo principal del cambio. Las razones que esgrimen los distintos Gobiernos son la independencia tecnológica frente a las empresas multinacionales, la seguridad que ofrece el SWL al permitir ver su código y el interés en desarrollar una industria informática local.

Se hace imprescindible conocer la situación actual que tienen algunos países de Latinoamérica en cuanto la conveniencia de adoptar el SWL como medio de garantizar la soberanía y como paso para su integración y liberación, viniendo de la necesidad de disminuir la brecha digital existente. Por eso, no se puede hablar de Cuba sin antes haber hecho un estudio global de cómo se manifiesta este sueño tecnológico para esos países, donde se tratan algunos aspectos importantes del impacto económico y avance que tiene este tipo de SW en Venezuela, Brasil, Argentina, México y Cuba por solo escoger estos, a pesar de que existen otros que también han avanzado en la lucha constante, que conlleva este gran reto.

2.1.1 Caso Venezuela con Decreto sobre SWL. Incremento del nivel de actividad en este campo

El planteamiento conceptual y el Modelo de Negocio que propone la corriente filosófica del software libre se conecta de una manera tan precisa con el proceso revolucionario venezolano, en torno al conocimiento y al uso de las TIC (Tecnologías de la Información y la Telecomunicaciones), porque entiende que el conocimiento producido a través del desarrollo de una aplicación tiene un valor concreto que es satisfecho cuando se paga su precio una sola vez. Esta valorización del conocimiento como producto único pretende impedir la explotación desmesurada y especulativa del conocimiento producido. [24]

La propuesta del software libre permite que el conocimiento producido pueda ser compartido y que sea disponible para que otros lo utilicen y mejoren, incrementando las potencias transformadoras del conocimiento.

El Gobierno Electrónico tiene como premisa fundamental la democratización del acceso a Internet y al Conocimiento en el marco del Plan Nacional de Tecnologías de Información y Comunicación, a través del cual se emprende el camino hacia el proceso de modernización del Estado, con la finalidad de proporcionar mayores y mejores servicios a la ciudadanía. Este Plan, es un instrumento que orienta la acción del Ejecutivo Nacional, con el propósito de establecer los lineamientos y políticas nacionales en materia de ciencia, tecnología e innovación. [25]

El desarrollo del Gobierno Electrónico en Venezuela, comenzó sus primeros pasos creando acceso a los servicios públicos, por medio de la Internet. Es así, como el estado obtiene el derecho a disponer de trámites y servicios que deben ofrecerse de manera gratuita, clara, sencilla, ágil, racional, pertinente y de fácil entendimiento. Con esto se persigue mejorar las relaciones de las personas con el gobierno y facilitar la eficaz y eficiente prestación de los servicios.

En la lucha por lograr la independencia, soberanía y la autonomía tecnológica, el presidente emitió el Decreto N° 3.390, (**ver Apéndice 1**) publicado en la Gaceta Oficial N° 38.095 de fecha 28/ 12/2004, donde todo material informático que compre la Administración o las empresas estatales debe llevar software libre. De esta forma, el Ejecutivo nacional establece que es prioridad del Estado incentivar y fomentar la producción de bienes y servicios para satisfacer las necesidades de la población, mediante el uso de

estas herramientas desarrolladas con estándares abiertos para robustecer la industria nacional, aumentando y aprovechando sus capacidades y fortaleciendo la soberanía venezolana. [26]

La puesta en marcha de este instrumento legal, por parte del presidente Chávez, faculta al Ministerio de Ciencia y Tecnología, para capacitar y formar a los funcionarios públicos en esta materia.

2.1.2 Brasil, Argentina y México

- **Brasil**

Brasil se ha convertido en los últimos años en uno de los principales exponentes a nivel mundial de la apuesta por el software libre como motor de cambio tecnológico y social. Numerosos proyectos de migración realizados en la administración pública, las universidades, y empresas privadas, permiten ahorrar dinero al mismo tiempo que se apuesta por la libertad y el futuro. Las iniciativas desarrolladas van desde la comercialización de PCs a bajo costo para llevar la tecnología al usuario doméstico, hasta la creación de telecentros de inclusión digital en los que los usuarios pueden realizar numerosas actividades. [27]

Dando un desafiante ejemplo a sus vecinos del MERCOSUR, el banco estatal brasileño “Banco do Brasil” abandonó el paquete de software Office de Microsoft y lo reemplazó con software libre para evitar el pago de derechos de licencia y dar así un paso adelante por lograr su soberanía tecnológica. [28]

Brasil consiguió desarrollar herramientas para la educación y la salud. No sólo para desarrollar software libre, sino también para usar lo que ya existía: mejorar y conseguir desarrollar los procesos para la sociedad manteniendo un proceso activo. Así como, herramientas de gestión de oficinas dentro del gobierno de Paraná, donde se desarrollaron sistemas importantes, estratégicos en el tema de la seguridad pública, gestión de automotores y otros... Se enfocó más en la parte económica, empresarial y de alto riesgo para la gestión gubernamental. Por esto el proyecto de Paraná está más bien concretizado que el del Río Grande del Sur. [29]

- **Argentina**

Ha pasado tiempo desde que se comenzó a hablar de la necesidad de utilizar software libre en Argentina. Obviamente, no es el único país donde se está impulsando el uso de herramientas de software libre para los sistemas del Estado. Hay varios casos de proyectos en Europa (Francia, España, Italia) y también en América (México, Brasil, Uruguay, Venezuela y otros).

En Argentina, el día 10 de septiembre del 2000 se presentó un Proyecto de ley sobre Software Libre en la Cámara de Diputados Nacional. Este proyecto de ley recibió una difusión aceptable y mucha gente se mostró a su favor. Actualmente el proyecto se encuentra en la Cámara de Diputados de la Nación. **[30]**

Es de esperarse que por la situación económica actual del país, se generen más casos en los que el software libre sea la única opción posible. Los precios de la tecnología se han multiplicado varias veces como consecuencia de la devaluación, pero como contrapartida de eso se abre la posibilidad para Argentina de exportar software. Sería muy provechoso que desde los sectores políticos se fomente este mercado y se lo impulsen para que crezca.

Lamentablemente y como sucede en estos casos donde los intereses económicos son tan importantes es de suponerse que las grandes empresas desarrolladoras y proveedoras de software para el estado continúen haciendo lobby para evitar el tratamiento del proyecto. **[idem, 30]**

- **México**

Aunque se conoce que el software libre ha tomado mucha fuerza en países como México también se sabe que aún no es considerado como una alternativa viable en la educación ya que el software propietario “invade” la mayoría de instituciones educativas en este país.

El presupuesto que México invierte en educación es poco alentador, si bien es sabido que México gasta mucho en educación, también se sabe que todo este presupuesto es más bien un mal gasto, a pesar del aumento de este destinado a la formación de los estudiantes de niveles básico, medio y medio-superior, la baja calidad de la educación ha sido un obstáculo al crecimiento.

En materia de estadística, sólo tres del tráfico generado por México en el uso de navegadores usa Firefox, de Mozilla, en comparación con países como Dinamarca, donde 45% lo emplea. La diferencia es enorme y habla de las posibilidades de crecimiento del software libre. [31]

Hay indicadores de que la situación está cambiando en el sector público, lo mismo sucede en las universidades, que deben hacer un trabajo más fuerte de formación en este tipo de herramientas.

El presidente lanzó un decreto de austeridad en el que se dice que “Las entidades paraestatales deberán promover la implantación de plataformas tecnológicas que representen ahorros netos en el mediano plazo”, eso es bien visto por parte de los partidarios del software libre, no tener que pagar necesariamente por licencias, y pone bajo total control de sus usuarios los equipos informáticos. [32]

La industria del software libre en México va a ser de las que van a tener mayor crecimiento; está siendo impulsada no solamente por empresarios y por empresas transnacionales, sino también por empresas Pymes, y en el gobierno pasa a ser ya una necesidad.

2.1.3 Significación del SWL en Cuba

Con lo expuesto anterior, de cómo estos países buscan estrategias, firman decretos y se imponen retos para conseguir lo que tanto anhelan, cabe destacar el papel que juega Cuba en este movimiento que está creciendo día a día, con sus dificultades y contratiempos.

Sin embargo, el Software Libre es sostenible en Cuba, a partir de las ventajas que tiene con respecto a los del tipo propietario. Su aplicación como plataforma informática de trabajo adquiere una relevante significación que puede verse desde ámbitos diferentes:

Económico: Su utilización no implica gastos adicionales por concepto de cambio de plataforma de software, por cuanto es operable en el mismo soporte de hardware con que cuenta el país. La adquisición de cualquiera de sus distributivas puede hacerse de forma gratuita, descargándolas directamente de Internet o en algunos casos a muy bajos precios, se garantiza su explotación con un mínimo de recursos,

por cuanto no hay que pagar absolutamente nada por su utilización (no requiere de licencia de uso, las cuales son generalmente muy caras), distribución y/o modificación. [33]

Tecnológico: Permite su adaptación a los contextos de aplicación, al contar con su código fuente, lo cual garantiza un mayor por ciento de efectividad, además la corrección de sus errores de programación y obtención de las actualizaciones y nuevas versiones. [idem, 33]

En el abril del 2004 se daba a conocer el Programa de Migración. Cuba daba los primeros pasos oficiales a nivel de gobierno para la migración al SWL, donde los tres grupos para el Programa son:

- Grupo técnico: es el que se encarga elaborar la guía de migración, desarrollar aplicaciones comunes, proponer y mantener el núcleo kernel, mantener los repositorios actualizados, entre otras.
- Grupo de capacitación: se encarga de elaborar los programas y cursos, organizar Jornadas Científicas y otros cursos a usuarios finales, y crear una cultura para el uso del SWL.
- Grupo legal: es el que le da seguimiento a los estándares abiertos, estudia y propone los aspectos legales y hace las propuestas bajo las licencias que se deben exportar.

Continúa desarrollando una serie de estrategias cuyo principal objetivo es la implantación del software libre y formar profesionales y técnicos informáticos en el uso del sistema operativo Linux como alternativa a la soberanía tecnológica, a través de los IPI (Instituto Politécnico de Informática), Joven Club de Computación y la Universidad de las Ciencias Informáticas (UCI), que son sin duda los centros con mayor aporte a esta causa.

2.2 Desarrollo del SWL en la UCI

La Universidad de las Ciencias Informáticas es una de las instituciones de Cuba con mayor contribución al SWL y que ha surgido al calor de la batalla de ideas, que impulsa el país.

La Comunidad de SWL de la UCI, nació en el año 2004 oficialmente. Antes de ese año hubo grupos de usuarios de GNU/Linux muy reducidos y sin contacto total o participación en la vida investigativa y tecnológica de la Universidad. En marzo del año 2005 apareció por primera vez el Portal de SWL y el primer repositorio de Debian.

Ya existía desde principios del curso 2004 -05 el Proyecto Nova, que tenía también su propio repositorio de Gentoo. Pero el surgimiento real de la comunidad y sus labores científicas y participativas se comenzaron a organizar a partir del curso 2005 – 06, con la creación del Proyecto UNICORNIOS que dio vida al portal de manera organizada, y en enero del 2006 ponía en funcionamiento el servidor de documentación libre, en febrero se efectuaba, organizado por el mismo proyecto, la migración total de los laboratorios docentes de la Facultad 10, iniciativa que había comenzado con el servicio de imágenes personalizadas para la docencia desde septiembre de ese curso. En ese mismo año uno de sus laboratorios estuvo en fase de prueba experimental con el sistema operativo Nova Linux, creado por estudiantes del Proyecto Nova de esta facultad.

Esta facultad tiene decenas de proyectos de investigación y producción, todos sobre software libre y con buenos resultados. Los líderes del Grupo de Investigación de Software Libre se han reunido con la comunidad de Software Libre de la UCI y han impartido conferencias, charlas, entre otros, con el objetivo de intercambiar ideas y conocimientos del tema.

La comunidad de SWL en conjunto con la FEU organizó el surgimiento de las demás comunidades y algunos de sus mejores miembros se clasifican hoy como coordinadores de comunidades de tecnologías libres. Actualmente, se trabaja en el Portal de SWL 2.0 (Zope/Plone) que tiene como objetivo final, certificar todas las acciones sobre el portal de SWL para ayudar en la formación de pregrado y postgrado de todos los miembros de su comunidad. **[34]**

2.2.1 Características y posibilidades de elaboración de SWL en la UCI

El desarrollo de SWL en la Universidad de las Ciencias Informáticas ayuda a incrementar el capital humano y a lograr la independencia tecnológica para el país. Es precisamente, tener este recurso humano informado, capacitado, comprometido con la patria, formando personal para este desarrollo, como una fuerza económica y científica, incluso política, y que señala un espacio de integración.

Por eso se está buscando basar el desarrollo en normas, en estándares, en un proceso industrial de desarrollo del SWL. Ahora se está tratando de tener relaciones con empresas nacionales e internacionales; que soliciten a la UCI la elaboración de un programa, para que las aplicaciones basadas en SWL se usen y sean generalizadas.

“Con las características que tiene la Universidad y también el país, nosotros utilizamos a veces software propietario. No se trata de una guerra contra este tipo de software, ni nada por el estilo. Se trata de lograr verdaderamente una soberanía. Hemos trabajado con algunos proyectos sobre Oracle, sobre MySQL y Postgret SQL, y la UCI ha ido implementando el uso del Postgret. También iremos trabajando durante este año en la formación de comunidades y grupos de desarrollo para completar la plataforma sobre Java y PHP. Y junto con eso ir creando toda la plataforma de desarrollo propia del país, que nos permita una verdadera soberanía tecnológica.” [35]

Todas las facultades que están inmersas en este proceso, no tienen el mismo nivel de avance, algunas sí basan algunos de sus proyectos en el desarrollo de aplicaciones libres, que en el **Anexo 3** se presentan. El proyecto de software libre, no es exclusivo de la UCI, pero sí se tiene una basta experiencia y una importante participación, principalmente en el grupo técnico de la comisión nacional y la creación de comunidades.

2.2.2 Disponibilidad actual de productos y servicios. Experiencia

En la UCI existen muchos proyectos de SWL, (ver Anexo 3) trabajando en la creación de un software o la evolución de alguno ya terminado. Los proyectos que están agrupados en el Polo “Software Libre”^{*} se dividen en 3 grupos:

- Sistema Operativo: NOVA.
- Plataforma de Productividad: RINDE-IFOR (herramientas para la gestión de proyectos, para desarrollar proyectos colaborativos y aumentar la productividad de los proyectos que tienen hosting allí).
- Migración y Soporte: UNICORNIOS.

Existen otras entidades dentro de la UCI, que trabajan en el desarrollo de productos para SWL como SOFTEL, y otros centros como CENTALAD que se dedican a proveer soluciones integrales y consultorías relacionadas con tecnologías de bases de datos y análisis de información.

Por ejemplo, la empresa **SOFTEL** actualmente está trabajando en la fase de desarrollo de un producto basado en software libre que abarcaría todas las áreas de gestión, adaptados al tema de un hospital o policlínico, este sería en un principio una aplicación Web y después sería hosting en un lugar para acceder a esa información desde un determinado sitio. [36]

Por otro lado está **CENTALAD** (Centro de Tecnologías de Almacenamiento y Análisis de Datos) que se encuentra desarrollando nuevas tecnologías de bases de datos, de procesamiento y representación de la información a partir del desarrollo de proyectos de I+D (enfoque a la soberanía tecnológica). Se caracteriza por la calidad de sus producciones y las publicaciones científicas de alto nivel que desarrolla. Entre los proyectos que se encuentran en desarrollo están:

^{*} EL polo de Software Libre es una unidad estratégica de desarrollo de la UCI, dedicado a desarrollar conocimiento, productos y servicios, con el fin de garantizar el proceso de migración a SWL y el uso de estándares abiertos, a nivel de Universidad Productiva, en el plano Nacional e Internacional.

- Proyecto: Generador de Reportes: MENPET de Venezuela, ERP Cubano, SIGE, Otros proyectos UCI.
- Proyecto: Sistema de Información para el INE de Venezuela.
- Proyecto: Sistema de Gestión de Datos de la ONE (SIGE).
- Proyecto: Desarrollo Almacén de Datos para DATACIMEX
- Proyecto: Postgret SQL este desarrolla un paquete de herramientas para el soporte y la extensión del gestor Postgret SQL que incluye el montaje de clúster de bases de datos sobre este gestor, soporte al diseño y la modelación de bases de datos, entre otras cosas. [37]

Entre los Productos y soluciones desarrolladas se pueden mencionar GEDERIX Generador Dinámico de Reportes Cedrux (figura 5), SIGE Sistema Integral para la Gestión Estadística Cedrux (Figura 6)*



Figura 5



Figura 6

También brindan una serie de servicios divididos en:

Servicios relacionados con Análisis Inteligente de Datos: en este tipo de servicios ofrecen servicios de integración y migración de datos, soporte a la integración con tecnologías, lenguajes y plataformas de bases de datos.

* Las figuras presentadas fueron obtenidas gracias a la colaboración de Dr. Pedro Yobanis Piñero Pérez.

Servicios relacionados con las Tecnologías de Bases de Datos: aquí brindan consultoría especializada en bases de datos, servicios de soporte, monitoreo y control de servidores y asesoramiento en Administración de Bases de Datos.

Servicios relacionados con la formación: desarrolla e imparte el perfil de formación pre graduada y post graduada en Tecnologías de Bases de Datos. [37]

Los trabajadores del centro poseen experiencia en el tema ya que han trabajado en el desarrollo de soluciones con aplicaciones reales como el desarrollo del sistema de Información Nacional de Seguridad Ciudadana (SINSEC) para el MPPRIJ en Venezuela, (BI & DWH); la creación del subsistema de generación de reportes (Estudio Psicosocial Pedagógico y Clínico-Genético de las Personas con Discapacidad “Misión José Gregorio Hernández”). Así como, el desarrollo de soluciones para el manejo de información y estadísticas donde cabe destacar el subsistema de generación de estadísticas e indicadores de información y la instalación de Salas Situacionales, Centros de Datos de altas Tecnologías y Estaciones de Trabajo; por solo mencionar algunos. [idem, 37]

➤ **Grupo de desarrollo Unicornios** (Servicios Especializados para la Migración a Software Libre): es el proyecto responsable de la migración, y asesora en esta materia e investigación de tecnologías libres al resto de las facultades, direcciones y proyectos de la UCI (Universidad de las Ciencias Informáticas). Se caracteriza por su apoyo a la Comunidad de Software Libre y a las necesidades de la migración de la UCI a software libre. Gracias a las experiencias que han tenido con países como Venezuela (al primero que le vendieron una Guía para la migración en el curso 2007-2008), Brasil, España e incluso Alemania, construyeron la guía personalizada, que actualmente están trabajando en la versión 0.32.

Entre los servicios que brinda se pueden mencionar: [idem, 7]

- UXI: Revista de Software Libre
- Sistema de Imágenes para la Docencia Facultad 10
- Portal de SWL
- Servicio de Documentación Libre

- Repositorio de Debian
- Concurso de Software Libre de la UCI

Cuenta con productos como **Infodrez v1.2** y **v2.0**, portales de Ajedrez basados en el CMS e107, **SistClon v0.3** es un sistema de clonación y distribución de imágenes, **RADIGRAX DVD** con personalización de Debian para diseñadores y modeladores de 3D.

Se desarrollan numerosos sistemas complejos asociados todos a alcanzar estados de migración:

Sistemas Web

- Plataforma de Migración a SWL
- Service Desk para la UCI (OneOrZero)
- Servicios con Minería Web
- Dotproject MA-GMPR-UR2
- Web de Investigaciones Facultad 10
- N-Internos

Sistemas de Escritorio

- SistClon v0.4
- O2Pmigration
- Tocaroro – Desktop
- Glive

➤ **Proyecto Nova:** El Proyecto Nova provee una línea de productos y servicios de calidad orientados a usuarios en el área de las tecnologías informáticas o que hayan experimentado un proceso de migración de sistemas Microsoft Windows a sistemas GNU/Linux. Nova es un sistema operativo basado en tecnologías GNU/Linux para entornos de escritorio que permite realizar trabajos de oficina, reproducir archivos de música y video, navegar por la Web, ver fotografías, entre otras aplicaciones, en respuesta a la necesidad de reducir los gastos por concepto de licencias de software y conduce a lograr un estatus de independencia y soberanía tecnológica.

Nova ofrece competencias de programación, versión del sistema diseñada especialmente para los torneos de computación de la ACM. Contiene los compiladores y herramientas de desarrollo necesarios para la exitosa realización de dichos certámenes.

Para terminales sin disco, tiene personalización diseñada para su uso en tecnología “diskless”. Está orientada a entornos de escritorio comunes permitiendo las mismas funcionalidades de Nova, pero con opciones de optimización para hardware de bajos recursos. Para la docencia, cuenta con una versión creada para los laboratorios docentes de la UCI. Contiene todo el software necesario para que los estudiantes reciban todas las asignaturas de la Carrera de Ingeniería en Ciencias Informáticas utilizando herramientas y tecnologías de SWL y SCA. [38]

Servicios de Nova:

- Soporte técnico
- Capacitación
- Creación de personalizaciones del sistema a la medida del cliente

A través de **Nova Desk** se brinda ayuda y soporte técnico a Nova que ofrece también, otros servicios como foros, noticias y blogs. Actualmente trabaja con:

- Summon, interfaz gráfica para el gestor de paquetes. **Figura 9**
- Serere, instalador de la distribución. **Figura 7**
- Capoeira, asistente para compartir carpetas utilizando Samba. **Figura 11**
- Ecumenix, asistente para unir Nova a Dominios Activos de Microsoft.
- Guano, un entorno de escritorio ligero para maquinas de bajas prestaciones. **Figura 10**



Figura 7: SERERE –Framework para Instalación

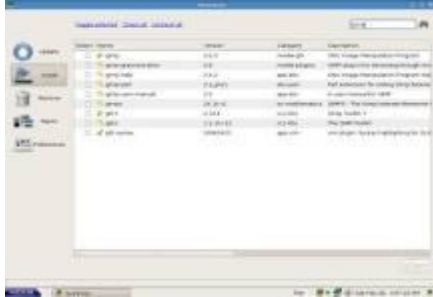


Figura 8: Centro de Control de Nova



Figura 9: SUMMON – EQUO – ENTROPY

Arquitectura y Sistemas para la
Instalación de Paquetes

Figura 10: GUANO

(Entorno de Escritorio Liger y Usable)



Figura 11: Capoeira – Ecumenix Integración con Redes y Dominios Windows

En el mes de febrero, se lanzó la distribución de NOVA, en La Habana durante la Convención Internacional “Informática 2009”, bajo el lema “Nuevas tecnologías: desarrollo y soberanía” con una amplia repercusión mundial.

El incremento de los proyectos que se desarrollan en software libre es cada día mayor, por lo que se pueden encontrar muchos desarrollos basados en PHP, Java y Python.

- En la Universidad existe la **Comunidad de SWL** que brinda servicios como:

- **Portal de SWL:** Noticias, Foros, FAQ, Mensajería Privada
- **Servicio de documentación libre:** UPLOAD, Actualizaciones
- **Repositorios:** Debian, Gentoo, Ubuntu, Free BSD, SUSE y Arch Linux
- **Wikipedia UCI**
- **Imágenes de GNU/ Linux**
- **Revista de SWL:** UXI
- **Servicio de Imágenes para la Docencia**
- **Concursos de SWL**
- **gforge UCI;** aún sin autenticación con LDAP

Otros servicios que se encuentran actualmente en Desarrollo:

- **RSS** para las noticias del Portal.
- **Certificación** de pregrado y postgrado.
- **Radio SWL**
- Un **Sistema de Información Geográfica** (SIG o GIS) para la residencia, como parte de las líneas de investigación en la UCI, sobre SWL.

Cabe señalar, que el grado de participación de los desarrolladores, de los Proyectos Productivos de la UCI que usan SWL en sus productos/servicios (**ver Anexo 6**), es de un 51,9% lo que significa que es un 3,8% mayor que la mitad de los que no están vinculados, ni participan en ninguna comunidad. Entre las comunidades a las que más se vinculan están:

1. Comunidad de SWL de la UCI
2. Comunidad de Drupal Hispano
3. Comunidad de Postgret SQL
4. Comunidad de Ingeniería de SWL
5. Comunidad de DotProject
6. Comunidad de Inteligencia Artificial
7. Comunidades de Desarrollo
8. Comunidad de Diseño Gráfico y Multimedia

Integrándose a estas de diferentes maneras como: a través de los foros, del propio proyecto, encuentros, portal de la comunidad, convocatorias, por motivación propia del desarrollador, y contribuyendo en gran medida con aportes de ideas, opiniones, desarrollos, respondiendo preguntas, solucionando errores y corrigiéndolos, investigando, difundiendo el SWL a otros países, posteando en el foro, elaborando documentos y videos editados en SWL.

Estas contribuciones por parte de los integrantes, de estos proyectos, han sido reconocidas en solo un 35%, lo cual es muy bajo, por lo que es recomendable incentivar a todos los miembros a continuar esforzándose en estos aportes, así como exhortar a los que no participan de ninguna forma a vincularse y sacar provecho de sus conocimientos de forma voluntaria. Esto puede generar comentarios inapropiados por parte de otras entidades y desarrolladores, perjudicando en el mercado la imagen y el prestigio de la universidad. Por eso, es muy importante mantener la presencia de la UCI en estas comunidades, como sustento de la credibilidad a la hora de ofrecer los servicios basados en estas plataformas a los clientes potenciales.

2.2.3 Formas de comercialización de SWL utilizadas hasta ahora. Ventajas y limitaciones.

En la UCI la comercialización de productos, soluciones integrales (que integran varios componentes) y los desarrollos con las instituciones, se realiza a través de ALBET, todo lo que ellos hacen es sobre la base de diferentes esquemas de comercialización, y el paquete de comercialización para un producto se está comenzando a aplicar ahora. Comercializa principalmente con entidades de Venezuela, y con otros países como Argentina, España, Canadá, entre otros.

ALBET utiliza varias formas de comercialización como el de **consultoría** que recoge todo lo que se realiza con los diferentes componentes de las soluciones integrales. “Nosotros trabajamos mucho la parte de diseño y la transformación organizacional, así como asesorías especializadas y componentes de desarrollo de software.” [39] Otro modelo que utilizan es el de comercialización de **tangibles** (bienes); que puede ser equipamiento para montaje de centro de datos, de redes, ofimáticos (impresora, computadora,

laptop) y productos de software, que pueden ser licencias (Oracle: para instituciones, el mismo Windows, entre otros.) o pueden ser aplicaciones más específicas (como un ERP). Todo esto constituye una solución integral con diferentes componentes.

“Esta es la manera que seguimos para vender la solución como tal, pero se puede vender un producto solo; como el **“ALAS PACS”**– que es un sistema de gestión de imágenes médicas (maneja las imágenes de los equipos de radiologías, tiene visualización para diagnóstico y transmite imágenes en determinados estándares)-. Otro producto es el **“ALAS HIS”**– es un sistema de gestión hospitalaria.”

ALBET también utiliza en el modelo del **HIS** y del **PACS** una venta por licencia, que es lo que hace Oracle y las grandes transnacionales del software, donde se le ofrece al cliente usar el **PACS** y por cada estación de visualización o de transmisión en que se use, se cobra una cierta cantidad de dinero.

Otro modelo que están explotando ahora, es el de un **“desarrollo conjunto con una entidad”**, el cual se usa fundamentalmente para los nuevos mercados. Cuando ellos entran en un nuevo mercado tratan de buscar aliados en esos países, y con éstos se aplica el modelo. “Nosotros tenemos experiencias de contratos de este tipo; nos dan anticipo y también ponen esfuerzo de desarrollo, conformamos un producto, nos dividimos la participación de este producto y salimos a venderlo. Si lo vendemos nosotros, pues es esta participación más un margen comercial que se fíe, si lo vende la entidad extranjera es el margen comercial (que se lo ganó él porque es el que está vendiendo) y la participación en función del esfuerzo que se hizo durante el desarrollo. Esa entidad extranjera puede darnos anticipos, pero si no los da entonces nosotros ponemos el riesgo de desarrollo y él, el riesgo que tiene que dar para ese desarrollo y se vende; entonces el % de participación sería diferente, y a la hora de la venta se aplica lo que son las **regalías**: Aquí hay diferentes modelos, porque no es que cada vez que vendamos un producto haya que transferir la cantidad; para ello se hacen conciliaciones semestrales con la entidad asociada; donde se muestran los estados de resultados y se demuestra cuanto se vendió y luego la entidad que vendió procede a transferir las ganancias a la otra.” [idem, 39]

Sin embargo, han adoptado el término de **“Solución Tecnológica Integral” (STI)**, en la que se le ofrece al cliente una STI con transferencia tecnológica incluida, para que se sienta más seguro y se decida a comprarlo, bajo un precio, que se puede conformar siguiendo las condiciones generales del comercio

Cuba-Venezuela, que se realiza anualmente. “Pero con PDVSA (que es una entidad con la que trabajamos) se conforma de acuerdo a unas determinadas tarifas, es decir, puede tener diferentes formas de conformación del precio, pero al final es una solución integral que tiene varios ítems que nosotros preciamos de una forma, y nos ocupamos de absolutamente todo para que ese cliente tenga toda esa solución como debe ser y la opere, y se le hace una transferencia de la solución.” [idem, 39]

Existen proyectos que se destacan por su rentabilidad y comercialización como los desarrollados para el CNTI, MINPPAL y PDVSA en Venezuela, ente otros¹³:

FILPACON es un software para el filtrado de contenidos en la web que permite regular (permitir o denegar) el acceso de usuarios a determinados contenidos de Internet. Para el proceso de filtrado utiliza, principalmente, una Base de Datos de URLs Categorizadas y la Política de Uso Aceptable de Internet definida por la organización. Cuando le vendió el producto de filtrado de contenido a CNTI, obtuvo ganancias de 124 000 USD y el costo fue menor de un 10 % por el desarrollo de este, el servicio que se realizó fue de instalación personal y entrenamiento (se dieron cursos de FILPACON como aplicación Web y de Apache como servidor), pero esto no constituye un plan oficial de servicios prestados. La visión del proyecto como tal es extenderse a su comercialización, e impartir cursos de filtrado de contenido y otros de esta naturaleza. Se encuentra desarrollando un Motor de Categorización Inteligente de Contenido, el cual va actualizando la base de datos mientras navega por la red, donde los criterios de categorización aumentan de acuerdo a las tendencias actuales y de esta manera se amplía su radio de acción. Cuando el producto esté terminado su servicio estrella estará enfocado, a un servicio de actualización, de desarrollo a pedidos a la medida del cliente que puede ser brindada de forma independiente o unida a estos servicios.

El **Grupo de desarrollo UNICORNIOS** trabajó en cómo migrar al SWL, un conocimiento que se usa para un servicio de consultoría, y ese servicio se vendió por primera vez el curso pasado (2007-08), a MINPPAL en Venezuela, donde se le trazó la estrategia completa de migración, se le dio un documento tutorial (contaba con más de 100 paginas); levantamiento de máquinas que tenían, cuales eran sus

¹³ El estimado del costo en los convenios con este país se calcula como horas de trabajo por categoría del especialista.

opciones y total de servidores, y agregado a eso se dieron cursos de capacitación. La ganancia neta de este negocio fue aproximadamente de 150 000 USD. El negocio como tal pudo haber sido más completo, con cierta continuidad, pero de las 4 partes que llevaba la migración sólo se realizó la primera. [40] Ha obtenido otros beneficios económicos a partir de los servicios que ofrece, principalmente en el soporte a la migración y los cursos de formación. Realiza una amplia actividad a nivel nacional, que aunque no se cobra, si puede valorizarse. Por ejemplo, los repositorios* que existen en la UCI como Debian, Ubuntu, entre otros, porque tienen una gran demanda ya que muchos clientes van a la Universidad en busca de descargas y actualizaciones. Se le ha vendido a ETECSA capacitación y a Fiscalía soporte para servidores.

Este año, UNICORNIOS está vendiendo soporte para servidores, conjuntamente con el Proyecto Hospitales de la Universidad, ellos hacen la aplicación que se va a instalar en el servidor, a la que hubo que hacerle modificaciones para que funcionara en un servidor Linux, porque Venezuela no admite licencias para servidores Windows. Por lo que ellos necesitan de especialistas que manejen las partes iniciales, como la elaboración de los cursos y de los paquetes de la aplicación elaborada por el Proyecto Hospitales y que se instale en los servidores y el manejo de los servidores.

El **Centro del Informática Industrial** desarrolla soluciones para la automatización, la optimización, la confiabilidad, y la simulación entre otras orientadas a su aplicación en frontera y la exportación. Estas se basan en las pericias y los resultados de innovación e investigación que se han consolidado en el país y se concretan en sistemas de supervisión para industrias, edificios y hospitales inteligentes, y productos y sistemas para la salud, el deporte, los juegos y la seguridad tiene como producto líder un configurador de SCADAs desarrollado para PDVSA. El Centro ha desarrollado soluciones para la automatización de procesos industriales en el área de la industria petrolera y especialmente para PDVSA. Más específicamente un sistema de supervisión y control de procesos que se encuentra en instalaciones de patio de tanques, casas de bombeo y pozos. Desarrolla en la actualidad sistemas de supervisión para otros sectores, aplicaciones para la instrumentación virtual, sistemas de supervisión para instalaciones de telecomunicaciones, sistemas para la supervisión de la industria azucarera y un sistema de confiabilidad.

* Las actualizaciones de los repositorios se hacen nacionalmente porque no hay mucho acceso a Internet.

Tiene un cliente externo (Mediante el Convenio Cuba-Venezuela y PDVSA-ALBET) y varios clientes nacionales (organismos y entidades del país). Informática 2009 sirvió para darse a conocer el proyecto, por lo que algunas entidades han comenzado a interesarse. Han firmado más de 15 contratos, tienen dos en análisis en estos momentos y existen alrededor de seis contratos nuevos. SCADA es más que un proyecto, es una organización que tributa como cliente fundamental a PDVSA.

Cabe destacar, El **Archivo General de la Nación de la República Bolivariana de Venezuela** (AGN) tiene la misión de preservar y difundir la documentación que representa la memoria histórica del país, adecuándolo a las crecientes exigencias del mismo y haciendo uso de estándares abiertos y de las Tecnologías de Información y Comunicación (TICs) para lograr la digitalización del Fondo Documental con el fin de evitar su manipulación directa y deterioro. Con vistas a desarrollar una solución de software a la medida para el Archivo General de la Nación de la República Bolivariana de Venezuela se desarrolló el software ArchiVenHIS (El sistema cuenta con varios niveles de acceso de usuarios de acuerdo al departamento o Área donde radiquen y las acciones que puedan ejecutar), el cual fue exportado a Venezuela (2006-2007), obteniendo excelentes resultados. En estos momentos se está negociando una nueva etapa de ese proyecto con Venezuela, pero además en Cuba se está trabajando con otros clientes desde 2008.

La comercialización de **NOVA** no pretende generar ingresos por su distribución como sistema operativo (SO) de Escritorio y la sustitución de importaciones representa beneficioso para el país. El Modelo de Negocio se basa en: Servicios de personalización de SO a la medida, incluido en soluciones integrales como plataforma base de los desarrollos UCI y embebido en las producciones de la industria nacional de hardware.

Entre sus alianzas se pueden mencionar:

- **Sabayon Linux** – Italia, Gestor de paquetes (y necesitan su Panel de Control, Instalador, Summon).
- **Ututo** – Argentina, Colaboración en desarrollo.
- **Funtoo** Código Fuentes, Resolución de errores.
- **Gentoo** - Colaboraciones indirectas

- **Canaima** – Planificación de Áreas Temáticas para el desarrollo de la Distribución del Alba.
- **MIC**, Grupo Nacional para la migración a SWL
- **Centro de Bases de Datos de la UCI.**
- **UCID**
- **Polos:** Polo de Gestión de la Información y Conocimiento, Facultad 10, Polo de Soluciones para Internet, Facultad 10, Polo de Telecomunicaciones, Facultad 2, Polo de Software Educativo, Facultad 8, Polo de Realidad Virtual, Facultad5
- **ETECSA**
- **ERP Cubano**

➤ **Ventajas**

1. **Ahorro:** desarrollar aplicaciones de SWL ayuda a ahorrar tiempo y dinero.

- **Bajo Costo:** el SWL no cuesta casi nada producirlo, excepto el tiempo, en caso de contratar alguna capacitación, no se paga un costo muy alto de licencias para empezar. Se produce un código propio, o se inventa un servicio a partir de uno, y se empieza a vender; es decir, nunca se empieza de cero, por lo cual éste aporta competitividad.
- **Menor Tiempo:** las aplicaciones desarrolladas sobre SWL parten de un código abierto inicial, esto ahorra tiempo, ya que resulta más trabajoso comenzar un proyecto desde cero. A lo mejor se invierte tiempo en una capacitación porque casi ningún personal del equipo de desarrollo domina Linux, pero si se cuenta con un grupo experimentado en la plataforma, en poco tiempo se logra la completitud del producto.

2. **Seguridad:** El producto implementado sobre una plataforma de libre como Linux le confiere una gran robustez en cuanto a los mecanismos de seguridad que implementa para su funcionamiento, ya que este Sistema Operativo es prácticamente inmune a casi todos los virus que existen en la actualidad.

3. **Calidad y actualización:** El SWL que se va a comercializar pasa por diferentes etapas para que de cierta forma se distribuya con calidad, esto es importante para la empresa que lo comercializa, que no

solo lo produce sino que ofrece una serie de servicios que están encaminados a resolver las necesidades del cliente, ya sea a través de la realización de nuevas versiones o herramientas de ayuda para el software libre.

4. **Independencia Tecnológica:** Al desarrollar una empresa su propio software bajo entornos libres se evita depender de los aportes o servicios que otras importantes empresas le puedan aportar.
5. **Innovación:** Se crean nuevos negocios y nuevas ofertas más baratas de comercializar.
6. **Progreso:** La posibilidad de crecer rápidamente y modificar los sistemas para adaptarse a condiciones cambiantes.

Generalmente se hacen productos multiplataforma para que su espectro de comercialización sea más amplio. O sea, produces sobre PHP, MySQL, Postgret, Java, C++, Phytion, y ya tienes tu producto estándar, luego se lo vendes a clientes que usen cualquier plataforma. [idem, 40]

➤ Limitaciones

A pesar de las ventajas del uso del SWL se debe tener en cuenta que la fase de maduración todavía dista mucho de la realidad actual, se pueden identificar algunas limitaciones que el desarrollador debe tener presente:

- **Desconocimiento de temas legales:** Existe desconocimiento por parte de los desarrolladores de la Universidad de los aspectos legales, como el derecho de autor, la propiedad intelectual, compatibilidad entre licencias y de los Modelos de Negocio existentes.
- **Propagación de la versión:** Un producto de SWL puede ser distribuido a través de una versión esta se prolifera en varias versiones que luego trae como consecuencia que para el usuario final se hace muy difícil a la hora de identificar la mejor versión del producto.

2.2.4 Importancia del tema para la UCI por promover los desarrollos en SWL

La necesidad de comercializar el SWL se está haciendo cada vez más importante, ya que de esta manera se pueden analizar con precisión la necesidad de regirse por un Modelo de Negocio. La comercialización

de SWL en la UCI no solo sirve para darle un mayor valor añadido al producto, sino también para lograr la independencia tecnológica del SW Propietario.

En términos prácticos, el empleo de un Modelo de Negocio basado en SWL beneficia a toda la universidad, debido a que:

- Contribuye a ahorrar tiempo, con lo cual se reducen los costos.
- Promueve la solidaridad y la colaboración entre las comunidades.
- Contribuye a la independencia tecnológica.
- Ayuda a mejorar la imagen comercial de la universidad siempre y cuando el producto tenga éxito.

Es bueno destacar que la empresa ALBET presenta un alto dominio del tema ya que es la encargada de llevar a cabo el proceso de comercialización que se realiza en la UCI, pero por parte de los desarrolladores de Proyectos Productivos basados en SWL, el 63% desconocen las formas de comercialización, así como de la aplicación de un Modelo de Negocio para sus productos/servicios.

2.3 Conclusiones

En este capítulo se pudo observar como los países Latinoamericanos se están uniendo a la lucha por el Software Libre para lograr su independencia tecnológica, tal es el caso de Brasil, Argentina, México, Venezuela, entre otros países que no fueron tratados en el capítulo y que no por eso son menos importantes en este tema. Esto ayudó al lograr un análisis general sobre la repercusión que tiene el SWL para adentrarse a las necesidades, debilidades y fortalezas que tiene Cuba en este campo y en especial la UCI como el punto central de esta investigación.

Esta universidad a pesar de la situación económica que enfrenta el país, traza estrategias para cumplir con una serie de contratos que satisfacen las necesidades del cliente, todo esto es logrado gracias a la experiencia con que cuenta la institución por parte de un grupo de profesores, ingenieros e incluso estudiante que investigan y se esfuerzan por dar lo mejor de sí a la hora de desarrollar un producto o brindar un servicio de SWL.

A través de una encuesta que se le realizó a diferentes desarrolladores de Proyectos que usan SWL, se supo que la participación de estos en las diferentes Comunidades y foros es baja, ya que la comercialización de los servicios depende en gran medida de las referencias y credibilidad del que los ofrece, para aportarle a la entidad un mayor prestigio a nivel internacional.

Se destacan varios proyectos en cuanto a la comercialización de sus productos y al nivel de sus desarrolladores aportándole a la UCI e incluso al país, un mayor prestigio y un sin número de ganancias, reconociéndolos como los más rentables y factibles de la universidad.

De acuerdo a las ventajas que brinda la comercialización del SWL, los profesionales pueden analizar a fondo los problemas y mejorar los desarrollos en todos los casos que sea necesario, nutriéndose del Software Libre, disponible globalmente bajo distintas licencias. Constituye un campo ideal para aplicar la creatividad e innovación, aspecto en el que los jóvenes universitarios alcanzarían buenos desempeños.

CAPÍTULO 3. TRES PROPUESTAS PRINCIPALES DE MODELOS DE NEGOCIO PARA EL SWL A UTILIZAR EN LA UCI.

3.1 El punto de partida

A pesar del poco tiempo que lleva ocupando a todos los involucrados el Software Libre en la universidad, se han logrado considerables avances de reconocimiento internacional que proyectan una fuente sustentable. Actualmente, la comercialización del Software Libre en la UCI se rige por Formas de Comercialización, que además de ser escasos, no explotan suficientemente todas las vías factibles para sacar un mejor provecho de los recursos materiales y humanos, para rentabilizar el conocimiento de los desarrolladores que poseen mayor nivel profesional y para seleccionar su público objetivo¹⁴, en general.

A través de un análisis coherente de los Modelos de Negocio identificados en la búsqueda bibliográfica, de las fichas conformadas en el Anexo 1 y las formas de comercialización del SWL en la universidad, se elaboraron una serie de fichas, ordenadas alfabéticamente de la A-J, que pueden ser aplicables a cualquier proyecto de la UCI. Y que serán tomadas como punto de partida en la realización de tres propuestas principales de Modelos de Negocio para la comercialización del SWL en dicha entidad.

Modelos que pueden ser aplicados en la UCI

A

Modelo: Comercialización con dos licencias.

Descripción: Consiste en asociar dos licencias a un determinado código, una de ellas es libre y la otra propietaria. El cliente puede elegir el tipo de licencia con las condiciones de una licencia de código abierto, como la de tipo BSD o la GPL o también puede elegir la licencia de código propietario.

Ventajas: La licencia GPL, permite crear múltiples niveles de sublicencias en forma de código abierto (normalmente bajo idénticas condiciones) y las licencias tipo BSD es posible re-licenciar el código derivado a cualquier tipo de licencia, incluidas licencias propietarias.

Desventajas: La licencia de código propietario limita los privilegios para re-licenciar el producto y lo establece como un estándar privativo.

¹⁴ Público objetivo: a quien va dirigido el producto.

Comentario: Este modelo estimula la introducción del software de código abierto en los negocios mientras estimula su desarrollo con los ingresos que brinda la venta de licencias propietarias.

B

Modelo: Comercialización de servicios asociados a un producto.

Descripción: Consiste en desarrollar una determinada aplicación bajo licencia libre, ofreciendo personalizaciones y/o servicios específicos sobre ese software. Por ejemplo, una empresa que distribuya el software libre puede crear su propia versión y ofrecer servicios añadidos, como soporte, personalización, consultoría o formación, control de riesgo y seguridad, entre otros, con los que hacen el negocio.

Ventajas: Las empresas pueden aportar un mayor valor añadido y aumenta sus márgenes operativos. Es una alternativa viable cuando se ha generado la necesidad de usar un software o servicio.

Desventajas: Los ingresos son a mediano plazo por las previas inversiones que se deben asumir. La calidad y la utilidad del software deben ser óptimas porque sus ingresos dependen de estos.

Comentario: Ofrecer estos servicios asociados a un determinado producto es importante pero se necesita de un personal calificado para brindar este servicio a los usuarios, a pesar de que se debe invertir al principio y de que los ingresos no son inmediatos, tiene grandes beneficios principalmente de los que pagan por un servicio más profesional.

C

Modelo: Comercialización de servicios integrales.

Descripción: Consiste en ofrecer servicios profesionales a usuarios interesados con este servicio. Entre los servicios que se pueden comercializar son entrenamiento y certificación, algún tipo de formación o consultoría por parte de la entidad.

Ventajas: Aporta un mayor valor añadido a la entidad. Brindar estos servicios profesionales ayuda a los ingenieros a obtener una mayor preparación

Desventajas: Se necesita de una buena preparación por parte de los ingenieros de la entidad a la hora de ofrecer los servicios.

Comentario: Este modelo no está enfocado en brindar servicios asociados a un producto sino ofrecer servicios de forma general, para ayudar al cliente a prepararse mejor en lo que es de su interés.

D

Modelo: Desarrollo de componentes comerciales.

Descripción: Se basa en el desarrollo bajo licencia libre de aplicaciones específicas, con un objetivo específico, en las que se desarrollan determinados componentes comerciales que cubren determinadas necesidades de un cliente. Esto puede ser a través de paquetes compuestos de un conjunto de componentes probados e integrados.

Ventajas: Las empresas pueden aumentar sus márgenes y competir en el mercado dónde sus posiciones son más ventajosas. Es posible construir aplicaciones acorde a las necesidades del cliente mediante la integración de los elementos.

Desventajas: Deben hacer grandes inversiones para la migración de sus aplicaciones a SWL y la formación de sus técnicos.

Comentario: Es un modelo horizontal donde estas empresas para poderse basar en código abierto deben migrar primeramente para poder independizarse de las empresas

E

Modelo: Comercialización de SWL con limitaciones.

Descripción: Este modelo consiste en vender el producto de SWL a un precio mínimo ya sea vendiendo la distribución propietaria durante un tiempo y luego libre donde pasado un tiempo (normalmente, cuando se empieza a comercializar una nueva versión, también como software propietario), esa versión pasa a distribuirse con una licencia libre; o limitar la distribución durante un tiempo, aquí el software es libre desde que el productor lo comienza a distribuir durante un tiempo sólo a sus clientes, que le pagan por ello (normalmente en forma de contrato de mantenimiento) y luego lo publica de manera gratuita.

Ventajas: Para la primera limitación la empresa productora obtiene ingresos de los clientes interesados en disponer lo antes posible de nuevas versiones, y a la vez minimiza la competencia, ya que cualquier empresa que quiera competir usando ese producto sólo podrá hacerlo con la versión libre. Y para la otra el productor obtiene ingresos de sus clientes, que perciben esta disposición preferente del software como un valor añadido.

Desventajas: Debido al retraso con el que el producto está disponible para la comunidad del software libre, es prácticamente imposible que ésta pueda colaborar en su desarrollo, por lo que el productor se beneficia muy poco de contribuciones externas.

Comentario: El modelo sólo funciona si los clientes a su vez no hacen público el programa cuando lo reciben. Se puede aplicar de diversas formas por ejemplo proporcionando el código fuente a sus clientes junto con los binarios, y cobrándoles por acceder al paquete formado por el código fuente, los binarios y los servicios profesionales. También, vendiendo el software en un CD junto a un manual de usuario, a un precio módico, a muchos le sería más fácil adquirirlo de esta manera que pasarse horas bajando el código.

E

Modelo: Elaboración de sistemas a pedidos.

Descripción: Este modelo consiste en realizar aplicaciones soportadas y desarrolladas sobre SWL, a partir de que una empresa determinada o usuario haga una solicitud de pedido al desarrollador del producto, de acuerdo a sus necesidades.

Ventajas: Se pueden construir aplicaciones a la medida del cliente mediante la integración de los elementos que más se adapten a sus necesidades. Permite al cliente modificar el código.

Desventajas: Como es a pedido se debe esperar por alguna solicitud.

Comentario: Con este modelo el proveedor se evita inversiones innecesarias y puede ofrecer además, soporte y servicios asociados con el producto. Si existe mucha demanda de productos el nivel de ingresos sería mayor.

G

Modelo: Venta de accesorios.

Descripción: Consiste en vender productos tangibles de un producto, ya sea CD, libros, manuales de usuarios, entre otros accesorios.

Ventajas: La empresa se beneficia de la venta de estos accesorios como otro servicio más.

Desventajas: En algún momento estos accesorios pueden perder su valor cuando pasa un tiempo determinado.

Comentario: Este modelo puede generar ingresos por varios métodos, donde siempre va existir un cliente que necesite estos servicios.

H

Modelo: Venta de SWL por imagen de marcas.

Descripción: Es utilizado por empresas que invierten en establecer una marca con buena imagen, y bien reconocida, que más adelante les permita vender con suficiente margen productos libres.

Ventajas: Puede ofrece una gran cantidad de servicios, tratando de rentabilizar el máximo posible esa imagen de marca

Desventajas: Las empresas tienen que invertir en publicidad.

Comentario: Aunque pueden conseguirse productos muy similares por menos dinero, muchos clientes están dispuestos a pagar el extra por comprar una marca. Este modelo se aplica generalmente cuando la entidad que libera el software es una empresa que cuenta con la propiedad intelectual del software y la marca.

I

Modelo: Hardware con SWL incluido.

Descripción: Se basa en software libre incluido en Hardware, mediante la adopción de estándares, de un Kernel estable y de nuevas tecnologías soportadas.

Ventajas: La combinación de SWL con un hardware específico es también una buena oportunidad para los editores de SWL para crear nuevos productos.

Desventajas: En ocasiones muchas empresas que se dedican a esto necesitan asociarse a otra que se dedique a esta parte del hardware, ya que ellas no están adentradas en este campo y instalar el SWL en el para luego venderlo.

Comentario: Los desarrolladores pueden integrar todos los componentes teniendo en cuenta los requisitos específicos de los sistemas del hardware, al trabajar con plataformas en entornos libres.

J

Modelo: Software propietario sobre plataforma de SWL.

Descripción: Este modelo consiste en construir software propietario utilizando herramientas libres o sobre una plataforma libre, y después vender sus servicios, a través de Internet.

Ventajas: Estos se han desarrollado gracias a las plataformas libres que hacen posible elaborar productos sin invertir en costosas plataformas propietarias, ni tener que realizar complicados desarrollos desde su inicio.

Desventajas: Este modelo no logra captar las ventajas competitivas derivadas de Modelos de Negocio puramente basados en software libre por lo que pueden llegar a tener dificultades para obtener rentabilidad a largo plazo.

Comentario: Existen aplicaciones que necesitan un gran número de usuarios en el mundo del software libre para poder obtener sus ingresos.

3.2 Formato de Modelo de Negocio

En la UCI la comercialización del Software Libre pretende aportar al mercado un Modelo de Negocio que se estructura a partir de los productos y/o servicios; es decir, se paga por los servicios adicionales que supone la adquisición del programa, la implementación de la plataforma, la capacitación del personal y el soporte técnico. El nuevo paradigma consiste en ofrecerlos en base a ese software, a través de la formación, instalación, mantenimiento, nuevos desarrollos, auditorías y otros; que para el cliente es un gran ahorro, y una buena elección.

Cuando una entidad comercializa el SWL tiene que tener en cuenta, si este cumple con las necesidades y requerimientos de ese nuevo mercado al cual va dirigido su producto/servicio, si puede ofrecer un soporte de calidad suficiente para ser atractivo, si habla su idioma para traducir su producto y si está al nivel de sus estándares de calidad.

Por eso, existe la necesidad de exportar SWL con el objetivo de obtener ganancias y alcanzar un prestigio internacional, a partir de las siguientes reglas básicas a seguir:

1. Las ventas se realizarán siempre bajo Licencia, que respetando los preceptos del SWL, limiten el traspaso a terceros sin el consentimiento o conocimiento de ALBET.

2. En el contrato de venta se tratará de incluir el soporte técnico por un período de un año. De no lograrse deberá estar listo para su contratación posterior.
3. Para el cumplimiento del contrato se organizará en el Polo Productivo un Equipo de Proyecto o se incorporará en uno afín ya existente.
4. El Equipo de Proyecto deberá garantizar la evolución del producto y la disponibilidad de una nueva Versión mejorada y con funciones adicionales en un plazo no mayor de un año.
5. Para cumplir el acápite anterior es necesario evaluar y decidir con un criterio comercial las funciones y opciones a entregar en cada versión.
6. El Equipo de Proyecto conformará el expediente completo de los productos y del cumplimiento de los contratos, teniendo en cuenta la normal rotación de estudiantes y eventual de profesores.
7. Desde las primeras etapas de la definición o negociación de un producto, deberán identificarse otros potenciales clientes y realizar una actividad de promoción.
8. El objetivo fundamental de esta estrategia comercial será lograr la venta de productos similares a diferentes clientes.

Estas reglas son importantes a la hora de aplicar un Modelo de Negocio eficiente y factible, que va acompañado a la experiencia que tiene ALBET en el tema de la comercialización principalmente con Venezuela –gracias al convenio intergubernamental existente - donde el esquema general, es que el elaborador dispone de un producto y/o servicio que le solicitan desarrollarlo y brindarlo en una temática de su competencia.

No hay dudas, de que los proyectos productivos de SWL en la UCI muestran un gran avance en su desarrollo, por lo que una vez definidas las formas de comercialización y las reglas básicas a seguir, se procede a conformar el formato de Modelo de Negocio¹⁵ siguiente:

Formato de Modelo de Negocio para SWL

¹⁵ **NOTA:** Es un formato indicativo, no obligatorio y ajustable según el caso.

- **Título** (nombre del modelo)
- **Descripción. Características principales** (descripción del modelo y sus características)

Sobre el producto y servicios

- **Cartera de productos/servicios** (productos ó servicio al cual le va aplicar el modelo)
- **Diferenciación de la competencia** (¿qué hace lo diferente a los demás?)
 - **Actividades de desarrollo**
- **Alianzas** (clientes, asociaciones con los que va a negociar)
- **Estrategia de desarrollo** (visión del proyecto, lo que quiere lograr con el modelo en el futuro)
- **Especialización. Líneas de I+D+i** (cómo se especializan en el tema, ya sea a través de la investigación u análisis de información que luego será aplicada a lo que se va a desarrollar)
- **Evolución y sostenibilidad** (¿cómo evoluciona el producto, para mantenerse?)
- **Rentabilizar el conocimiento** (¿cómo le sacas provecho al conocimiento del personal que posee un alto nivel?)
 - **Recursos necesarios** (recursos que necesitas para brindar el servicio)
 - **Personal. Capacitación del personal propio** (personal que está disponible para esto y la capacitación que necesita)

Sobre el entorno (mercado (tendencias), competidores, distribuidores u otros factores, características de los países entorno legal, político, económico)

- **Relación con los Clientes** (¿cómo se relaciona (on-line, presencial)?)
- **Competencia** (mayores competidores)
- **Público objetivo** (a quién está orientado)
- **Formas de entrada al mercado** (por Internet, personalizado, ¿de qué manera?)
- **Relaciones con Comunidades y Foros** (si está vinculado a alguna comunidad o foro, juega con la estrategia de desarrollo)

- **Experiencia anterior.**
 - **Obtención de beneficios** (¿cómo obtienes los beneficios? y ¿por qué cobras (productos o servicios)? Puede ser la cantidad en pesos.)

ELEMENTOS QUE FUNDAMENTAN EL MODELO DE NEGOCIO

- **Ventajas** (del producto con respecto a otros)
- **Desventajas** (del producto con respecto a otros)

3.3 Tres propuestas principales aplicadas a tres Casos de Estudio

El formato anterior será utilizado en tres propuestas principales, para buscar diferentes vías viables y económicas a la hora de aplicar los diferentes modelos a algún proyecto productivo de la UCI, así como cualquier entidad del país que esté interesado en usarlo. Estas propuestas demostrarán al desarrollador, que el SWL puede ser comercializado a través de los productos y/o servicios, no sólo a nivel nacional, sino también internacional, lo que les ayudará en gran medida, a conocer como aplicar un Modelo de Negocio una vez que se piense en desarrollar una aplicación o se desee ofrecer un servicio determinado y no después que esta esté desarrollada, además de saber cuáles son los más ajustables y rentables de acuerdo a sus productos y servicios.

A continuación se aplican las tres propuestas principales, bajo un mismo formato, a tres Casos de Estudios que radican en la Universidad de las Ciencias Informáticas, seleccionados de acuerdo a los servicios que brindan y productos que ofrecen. Los proyectos productivos escogidos y las propuestas que se le aplican son:

1. Proyecto Unicornios se le aplica la Propuesta C: “Comercialización de servicios integrales.”

2. Proyecto Filpacon se le aplica la Propuesta B: “Comercialización de un producto y sus servicios asociados”.
3. Proyecto SCADA para PDVSA se le aplica la Propuesta F: “Desarrollo de sistemas a pedidos.”

3.3.1 UNICORNIOS

Propuesta C

- **Título:** Comercialización de servicios integrales.
- **Caso de Estudio:** Proyecto de Migración y Comunidad de Unicornios.
- **Descripción. Características principales.**

Consiste en ofrecer servicios profesionales asociados al conocimiento que tenga la entidad, siendo estos: entrenamiento y certificación, varios tipos de formación, consultoría y soporte técnico, a usuarios interesados en estos servicios. Por ejemplo una entidad que tenga un producto, no importa que lo haya obtenido por otra vía, y necesita especializarse en este, recurre a una entidad que posee el conocimiento especializado, en busca de una capacitación o formación de acuerdo a sus intereses.

Sobre el producto y servicios

- **Cartera de productos/servicios:** Relación de los productos y/o servicios con su descripción, condiciones y precios. Por ejemplo, en el Grupo de Desarrollo UNICORNIOS, el proyecto de Migración y Comunidad ha creado una estrategia para la migración al SWL donde uno de sus flujos de trabajos está encaminado a brindar servicios de formación del personal y certificación del mismo por niveles de usuarios y este conocimiento está basada en la plataforma **GNU/LINUX Debian 4**. Durante toda la migración, ofrece este tipo de servicio, comenzando por la justificación del proceso y abarcando hasta las etapas posteriores al soporte, siendo el objetivo fundamental de la misma, la capacitación a todo el personal en los nuevos sistemas informáticos que se implantan en la institución.
- **Diferenciación de la competencia:** El conocimiento amplio en la temática de SWL y en específico en LINUX, forma parte de un equipo que desde hace más de un año, desarrolla productos, elabora

sistemas a pedidos y brinda una formación general y especializada en los temas de SWL, que le permite al proyecto disponer de especialistas calificados para brindar los servicios de consultoría y formación.

- **Actividades de desarrollo**

- **Alianzas:** Presta servicios a otras facultades de la Universidad de las Ciencias Informáticas, como el Proyecto Hospitales de la UCI para brindar soporte para servidores y con los Politécnicos de Informática de Cuba. También, con el equipo de desarrollo Fábrica de Portales, para una aplicación Web. Con entidades nacionales como: ETECSA (capacitación), Fiscalía (soporte para servidores).
- **Estrategia de desarrollo:** Este servicio tiene como propósito agrupar todas las ideas de los usuarios para desarrollar aplicaciones que tributen a ello y ampliar ofertas en el tema de migración, brindando otros cursos más avanzados y nuevos productos de SWL para los sistemas de Información Documental. Confeccionar e impartir planes de formación a los usuarios según el nivel de los mismos, donde se proponen tres niveles básicos de usuarios que se pueden identificar, esta propuesta puede desglosarse en más niveles en dependencia del lugar:
 - Formación de instructores en Software Libre: El objetivo de estos planes de formación es preparar un grupo de instructores que faciliten el proceso de formación de herramientas libres tanto del personal técnico como de los usuarios finales. Los instructores recibirán cursos prácticos y teóricos de las herramientas que serán utilizadas en los cursos de soporte técnico y usuarios finales.
 - Formación de soporte técnico: Los planes de formación orientados a este contingente tienen la finalidad de preparar un conjunto de personas para dar soporte a los usuarios cuando estos los necesiten a través de diversos métodos, estos usuarios recibirán un plan de formación que contendrá cursos prácticos y teóricos, así como que se les indicarán los mecanismos más adecuados para acceder a los recursos de Software Libre que les garanticen brindar soporte de excelencia.
 - Formación de los usuarios finales: La finalidad de estos planes de formación tienen como objetivo formar en Software Libre a los usuarios finales que usarán las aplicaciones instaladas en el nuevo sistema, para ello se prepararán de igual manera cursos prácticos y teóricos a la

medida de las herramientas utilizadas en este grupo pueden incorporarse secretarías, personal administrativos, algún número de estudiantes, entre otros. Se les debe dotar además las principales direcciones de los recursos donde estos puedan apoyarse en dependencia de los medios establecidos para dar soporte y capacitación.

- **Especialización. Líneas de I+D+i:** Después de alcanzar la migración de la Facultad 10 de la UCI en marzo del 2006 a software libre, el proyecto derivó en un grupo de **Investigación + Desarrollo**. Está compuesto por estudiantes de todos los años, que investigan tecnologías libres para diversas funcionalidades. Se desarrollan en él numerosos sistemas complejos asociados todos a alcanzar estados de migración más eficientes. Ahora se está trabajando en un modelo de negocio/comercialización para la producción de audiovisuales apoyados en SWL (mediante un Trabajo de Diploma de Pregrado), que sería algo novedoso para la universidad.
- **Evolución y sostenibilidad:** Es importante crear un mecanismo de certificación para certificar cada curso vencido e ir estableciendo determinados niveles en los usuarios para lograr una formación más adecuada, a través de cursos de capacitación. Al ir migrando las otras facultades de la UCI y todas las entidades del país al SWL, se le irán brindando a cada una los servicios de formación, así como soporte a los productos migrados por cierto período de tiempo.
- **Rentabilizar el conocimiento:** La forma de rentabilizar el conocimiento será mediante la incorporación de nuevos cursos de formación y mejorando los existentes. También, las experiencias en migración permiten obtener nuevos contratos y cumplirlos de forma más eficiente y con menos costo. Los estudiantes de 5to año dejan al graduarse un manual de desarrollo, por lo que el conocimiento se queda vigente para que los nuevos estudiantes que se incorporen al proyecto, puedan estudiarlo. Existe un grupo de soporte que confecciona Videos, Cursos y conferencias para la capacitación de otros.
- **Recursos necesarios:** Se necesita de un personal calificado y que domine bien todo lo relacionado con la migración, para esto es necesario contar con:
 - Crear canales para la atención a los usuarios vía telefónica ante dudas.
 - Habilitar algún local, laboratorio stand o espacio físico, con acceso a Internet, a donde puedan dirigirse los usuarios a recibir asesoría.

CAPÍTULO 3. TRES PROPUESTAS PRINCIPALES DE MODELOS DE NEGOCIO PARA EL SWL A UTILIZAR EN LA UCI.

- Participación en Eventos nacionales e Internacionales.
 - Participación en cursos en el extranjero
 - Posibilidad de suscripción a encuentros especializados y compra de libros, entre otros.
- **Personal. Capacitación del personal propio:** Se cuenta con 96 miembros del proyecto entre ellos: estudiantes, técnicos e ingenieros, que están bien preparados en el tema de la migración para brindar el servicio de formación y para darles capacitación a los nuevos miembros del propio proyecto, que lo necesiten.

Sobre el entorno (mercado (tendencias), competidores, distribuidores u otros factores, características de los países entorno legal, político, económico)

- **Relación con los Clientes:** Generalmente mediante contratos con los clientes, para una actividad específica o por un período de tiempo. On-line ofrece servicios de consultoría y formación y de forma presencial la de soporte técnico, y actualización de versiones o repositorios, en este caso los clientes van a Unicornios, donde se les descargan los repositorios para un disco y se les dejan los manuales para su posterior instalación.
- **Competencia:** En Argentina **Tucumán Linux**, es una empresa que se dedica a brindar todo tipo de servicios asociados a la plataforma libre de Linux, aunque Unicornios se destaca no solo por ofrecer esto que hace Tucumán, sino que brinda una gama de productos acompañados con sus servicios, así como otros asociados a los repositorios, al entrenamiento y capacitación de todo aquel que esté interesado en migrar a plataformas libres, cosa que esta empresa no tiene habilitado en la sección de servicios de su página Web. **iXsystems** que utilizan sus propios sistemas operativos como PC-BSD que en su página Web brindan una serie de servicios asociados a la migración, al soporte y a la venta de accesorios, esto implica que Unicornios se convierta en uno de los proyectos que más servicios ofrece. Otras como **AJ Informática** en Argentina que brinda Soporte Técnico de PCs a Domicilio, Instalación de Sistemas Operativos en PC y Notebooks, Auditoría de Redes / Seguridad WiFi, Migración de Sistemas Windows a GNU/Linux y Reparación y Mantenimiento de PCs esta es una empresa que pudiera servirle de apoyo a Unicornios en cuanto a todo a la integración que tienen de estos servicios.

CAPÍTULO 3. TRES PROPUESTAS PRINCIPALES DE MODELOS DE NEGOCIO PARA EL SWL A UTILIZAR EN LA UCI.

- **Público objetivo:** Está orientado a empresas nacionales que se dedican a la educación a la salud y otros sectores del estado, y a entidades internacionales, dedicadas a la educación, la salud, la industria y las telecomunicaciones.
- **Formas de entrada al mercado:** La salida al mercado es a través de ALBET. Los productos que lleva la Guía de Migración como desarrollo salen al mercado a través de ella. También, se da a conocer al mercado a través de NOVA, ya que es un grupo de desarrollo que apoya este proyecto.
- **Relaciones con Comunidades y Foros:** UNICORNIOS se caracteriza por su apoyo a la Comunidad de Software Libre de manera sistemática, y a las necesidades de la migración en la UCI a software libre. Gestionó el Portal de Software Libre de la UCI, organizando la comunidad y los servicios de dicho portal como: Servicio de Foros, Noticias, Encuestas, FAQ y Gestión de usuarios, aportando de forma libre los conocimientos por la vía de los Foros.
- **Experiencia anterior.**
 - **Obtención de beneficios:** UNICORNIOS obtiene beneficios económicos a partir de los servicios que ofrece, principalmente en el soporte a la migración y los cursos de formación. Realiza una amplia actividad a nivel nacional, que aunque no se cobra, si puede valorizarse. Ha cumplimentado contratos de exportación de aproximadamente de 150 000 USD, el curso pasado (2007-2008) con MINNPAL en Venezuela.

ELEMENTOS QUE FUNDAMENTAN EL MODELO DE NEGOCIO

- **Ventajas:** Muy pocas empresas a nivel mundial dedicadas al SWL están especializadas en el tema de la migración (Tucumán Linux) en diferentes plataformas libres. Unicornios posee un personal que colabora con otras comunidades de forma cooperativa y desinteresada, mientras que otras utilizan sus recursos humanos para competir en el mercado (del producto con respecto a otros).
- **Desventajas:** La certificación de los profesionales y el poco reconocimiento a nivel internacional. Falta de recursos tecnológicos que faciliten el desarrollo de componentes basados en SWL y la comunicación con los clientes.

3.3.2 FILPACON

Propuesta B

- **Título:** Comercialización de un producto y sus servicios asociados.
- **Caso de estudio:** Proyecto Filpacon.
- **Descripción. Características principales.**

Consiste en desarrollar una determinada aplicación bajo licencia libre, ofreciendo personalizaciones y/o servicios específicos sobre ese software. Por ejemplo, una empresa que distribuya el software libre puede crear su propia versión y ofrecer servicios añadidos, como soporte, personalización, consultoría o formación, control de riesgo y seguridad, entre otros, con los que hacen el negocio. Para usar este modelo se requiere de un personal altamente calificado (mejor si tiene certificación LPI) para que el cliente se sienta seguro a la hora de usar el servicio.

Sobre el producto y servicios

- **Cartera de productos/servicios:** Relación de los productos y/o servicios con su descripción, condiciones y precios. Por ejemplo, el Proyecto FILPACON es un software de código abierto, en su primera versión se distribuyó con licencia GPL, la tendencia en productos con este tipo de licencia es centrar sus beneficios en los servicios asociados como son: consultoría, actualización, capacitación y soporte técnico. Por lo que su segunda versión va enfocada a estos servicios e incluye como elemento novedoso un Motor de Categorización Inteligente de Contenido (se encuentra en desarrollo), que no va incluido en la venta de su aplicación Web de filtrado de contenido como producto, sino que permite prestar el servicio de actualización de las listas de URL prohibidas. Este motor tiene varios productos implicados para realizar una clasificación general de una URL, como son: un Detector de rostro, Detector de Imagen Desnuda y un Categorizador de Texto.
- **Diferenciación de la competencia:** Esta nueva versión permitirá disponer de un nuevo servicio distintivo y que aportaría los mayores beneficios para la entidad: es el de la actualización de las URL categorizadas a partir de un motor de inteligencia artificial que agiliza y hace más eficiente este proceso. Este servicio permitirá al cliente mantener el filtro actualizado ante nuevas páginas que

violen las políticas determinadas por lo que aumenta la eficiencia del filtrado. Este atributo permite sin dudas una eficacia sostenida del sistema evitando la violación de las políticas de seguridad de la entidad, adicionalmente se incorporan otros algoritmos de filtrado que le incorporan valor al producto y flexibilidad de cara al cliente. Son pocas las empresas que cuentan con este nivel de desarrollo en sus aplicaciones.

- **Actividades de desarrollo.**

- **Alianzas:** Joven Club de Cuba, Segurmática en la UCI, OSRI (Oficinas de Segurmática para Redes informáticas).
- **Estrategia de desarrollo:** Se pretende en un futuro impartir cursos de Filtrado de Contenido y otros de esta naturaleza, así como de ética informática debido a la importancia que estos traen consigo. Integrar los productos que están implicados al motor, mediante un módulo decisor de este; el cual realiza la categorización general mediante los resultados de los demás productos y distribuirlo bajo licencias de tipo BSD o MIT. Prevé brindar, un servicio de actualización, de desarrollo a pedidos a la medida del cliente y capacitación que puede ser brindada de forma independiente o unida a estos servicios.
- **Especialización. Líneas de I + D + i:** Se realizaron varias investigaciones y análisis del Estado del Arte, tomando lo mejor, de los demás filtros relacionados con la detección de rostros, imágenes desnudas para conformar, los productos que están implicados al motor. Se decidió incluir dentro de los servicios asociados un plan de cursos sobre ética informática y profesional.
- **Evolución y sostenibilidad:** el producto cuenta con un motor de categorización inteligente de contenido que va actualizando la base de datos mientras navega por la red, donde los criterios de categorización aumentan de acuerdo a las tendencias actuales y de esta manera se amplía su radio de acción. Cuando el producto esté terminado se creará un sitio para publicar las actualizaciones del motor, de manera libre, para que el usuario pueda adquirirlo y emitir los errores que este pueda tener, para que la en próxima actualización estos estén resueltos.
- **Rentabilizar el conocimiento:** La forma de rentabilizar el conocimiento será mediante la incorporación de nuevos cursos de capacitación y mejorando los existentes, ya que los miembros del

proyecto están especializados en temas de instalación, entrenamiento y consultoría, lo que representa parte de las ganancias a la hora de brindar sus servicios.

- **Recursos necesarios**

Cuenta con: 5 PCs, 1 PC para el servidor

Estos recursos tecnológicos no son suficientes ya que retrasa el trabajo del proyecto. Se necesita un laboratorio con varias máquinas conectadas a Internet, donde haya un teléfono. Participación en eventos internacionales y nacionales, en cursos de capacitación, entre otros.

- **Personal. Capacitación del personal propio.**

Se cuenta con un equipo capacitado para entrenar a nuevos miembros en temas de servicios asociados y familiarización con el software. A través de la participación en jornadas científicas, cursos de entrenamiento, entre otros, los estudiantes egresados dejan vigente su conocimiento para que los nuevos integrantes puedan utilizarlo.

Sobre el entorno (mercado (tendencias), competidores, distribuidores u otros factores, características de los países entorno legal, político, económico)

- **Público objetivo:** Orientado a instalarse en servidores para medianas redes de usuarios: colegios, pequeñas y medianas empresas, bibliotecas, cibercafé, entre otros. Aunque, existen varios países que pudieran convertirse en clientes potenciales teniendo en cuenta tendencias y relaciones político-comerciales que se han evidenciado, como son:
 - Venezuela
 - Brasil
 - Argentina
- **Relación con los Clientes:** Generalmente mediante contratos con los clientes, para una actividad específica o por un período de tiempo. Cuando Filpacon se instaló en los infoCentros de Venezuela, - de estructura homóloga a los Joven Club cubanos-, el servicio que se realizó fue de instalación

personal y entrenamiento (se dieron cursos de Filpacon como aplicación web y de Apache como servidor), pero esto no constituye un plan oficial de servicios prestados.

- **Competencia:** Entre sus competidores están: SQUIDGUARD, DANSGUARDIAN donde los recursos humanos de estas empresas están enfocados directamente en vender sus productos para competir en el mercado, mientras que Filpacon los utiliza de forma cooperativa y productiva. Estas empresas en cuanto a las líneas I+D, sus desarrollos están orientados al costo, cosa que no sucede con Filpacon ya que este es un proyecto que orienta sus diseños al resultado que pueda tener para luego adaptarlo según las necesidades del cliente. Existen otros filtros de contenidos como: El Servidor de **Web Segura Cyan** es de nivel de portador, un servidor filtrador proxy para Linux de alto rendimiento que tiene una avanzada base de datos URL, soporte de autenticación (LDAP y Directorio Activo), despliegue sencillo, y utilización remota, **Muro Activo Profesional (Active Wall Professional) 2.0** provee continua protección para toda la empresa y se destaca por la entrega comprensiva seguridad de red, permitiendo a los negocios proteger sus activos corporativos, así como **Suite 602LAN (602LAN SUITE) 2004.0.07** es un suite de programas de red todo en uno fácil de instalar y de usar, provee servicios de correo seguro, con anti-virus y anti-spam, soporta fax, muro de fuego construido con NAT y filtrado proxy de contenido web para compartir internet controlado. Combina el filtro antispam basado en contenido Bayesiano, listas negras y listas blancas juntas en un filtrado de anexos de correo para proveer completa protección correo basura.
- **Formas de entrada al mercado:** la única forma que han utilizado es mediante acuerdos entre CNTI y la Gerencia de Informatización de la UCI, para obtener el producto. Otra puede ser mediante la venta en la red o utilizando intermediarios.
- **Relaciones con Comunidades y Foros:** se relacionan con el Foro Framework contribuyendo al grupo de Google.
- **Experiencia anterior.**
- **Obtención de beneficios:** FILPACON cuando le vendió el producto de filtrado de contenido a CNTI, obtuvo ganancias de 124 000 USD y el costo fue menor de un 10 % por el desarrollo de este. Puede obtener beneficios económicos a partir de los servicios que ofrece, principalmente en el soporte al

motor que actualmente se encuentra en desarrollo. Realiza una amplia actividad a nivel nacional, que aunque no se cobra, si puede valorizarse.

ELEMENTOS QUE FUNDAMENTAN EL MODELO DE NEGOCIO

- **Ventajas:**
 - Con los actuales filtros a nivel mundial la minoría son sobre software libre.
 - Además aquellos que son libres y líderes en el mercado como (SQUIDGUARD, DANSGUARDIAN, estas crean un espacio para hablar de la competencia, así como para dar consultoría acerca de su instalación y soporte) y que no cuentan con una interfaz de administración tan completa como la de FILPACON.
 - El sistema de actualización a través de la Base de Datos es sumamente novedoso con el uso de un motor inteligente para la categorización de las URL.
 - Brinda la posibilidad de agregar o quitar funcionalidades, según las necesidades y requerimientos del cliente.
 - Tiene la posibilidad de crear listas de actualizaciones compatibles con otros filtros, buscadores, bases de datos de información, de forma tal que se le brinda el servicio a otras instituciones.
- **Desventajas:** La certificación de los profesionales y el poco reconocimiento a nivel internacional. Carencia de infraestructura tecnológica que facilite el desarrollo y el posterior soporte.

3.3.3 SCADA para PDVSA

Propuesta F

- **Título:** Desarrollo de sistemas a pedidos.
- **Caso de Estudio:** Proyecto SCADA para PDVSA.

- **Descripción. Características principales**

Este modelo consiste en realizar aplicaciones soportadas y desarrolladas sobre SWL, a partir de que una empresa determinada o usuario haga una solicitud de pedido al desarrollador del producto, de acuerdo a sus necesidades, a través de un contrato.

Sobre el producto y servicios

- **Cartera de productos/servicios:** El Centro del Informática Industrial desarrolla soluciones para la automatización, la optimización, la confiabilidad, y la simulación, entre otras, orientadas a su aplicación en frontera y la exportación. Estas se basan en las pericias y los resultados de innovación e investigación que se han consolidado en el país y se concretan en sistemas de supervisión para industrias, edificios y hospitales inteligentes, y productos y sistemas para la salud, el deporte, los juegos y la seguridad tiene como producto líder un configurador de SCADAs. Ha desarrollado soluciones para la automatización de procesos industriales en el área de la industria petrolera y especialmente para PDVSA. Más específicamente un **Sistema de supervisión y control de procesos** que se encuentra en instalaciones de patio de tanques, casas de bombeo y pozos. Desarrolla en la actualidad sistemas de supervisión para otros sectores, aplicaciones para la instrumentación virtual, sistemas de supervisión para instalaciones de telecomunicaciones, sistemas para la supervisión de la industria azucarera y un sistema de confiabilidad.
- **Diferenciación de la competencia:** La competitividad de los productos del centro se sustenta en la capacidad de integración de sus soluciones con muchos de los procesos de la entidad receptora, la rapidez de respuesta y la calidad de la atención a las necesidades del cliente, los relativos bajos niveles de inversión requeridos en sus proyectos, la producción sustentada en la reusabilidad de soluciones, componentes, bibliotecas, subsistemas que brinda la comunidad de software libre, una estructura que garantiza una alta eficiencia en el uso de los recursos humanos y que su recurso fundamental es la producción intelectual basados en el trabajo comprometido de su personal; desde los inicios del proyecto los plazos de entrega en los contratos eran cerrados y aun así fueron asumidos, lo que sirvió para ganar la confianza y satisfacción del cliente.

- **Actividades de desarrollo**

- **Alianzas:** CERCONI es una alianza en desarrollo, con el DST tienen algo parecido a una alianza que está sufriendo los embates de la crisis económica y la interpretación que se le está dando a ambos lados del Caribe, CEDAI con quienes sin formalizarlo han avanzado mucho
- **Estrategia de desarrollo:** Centro de desarrollo de soluciones para la automatización con un sistema maduro de innovación y producción, flexible, con el logro de haber insertado productos en las áreas de industria, edificios inteligentes, entretenimiento activo y preparación deportiva, seguridad de instalaciones y sistemas para la salud. La competitividad de sus productos y servicios, basados en los principios del Software Libre y la reusabilidad de componentes, combinados con una atención y soporte rápidos a las necesidades del usuario, el enfoque de las soluciones orientado a la integración de diversos e importantes procesos y su equipamiento en las entidades clientes, unido a otros factores como la innovación, la alianza estratégica con otras entidades -nacionales en primer lugar- y sobre todo el compromiso de su comunidad de desarrolladores e investigadores en sectores del mercado que garantizan un crecimiento sistemático de la organización y la adaptación a las complejas y cambiantes condiciones externas.
- **Especialización. Líneas de I+D+i:** existe una especialización profunda, porque el dominio de las aplicaciones en las que se enfrentan requiere de un conocimiento importante del Modelo de Negocio y la automática, las líneas también han sido una forma de especialización, es el camino para que los desarrolladores se especialicen y hagan I+D, es decir el proceso de búsqueda de investigación de desarrollo de prototipos y soluciones es fértil en este proyecto. En cuanto al tema de las publicaciones, la formalización de los resultados existe cierta debilidad, pues se carece de una estructura bien definida que se haga totalmente responsable de sus funcionalidades y tareas. Es una investigación orientada a la innovación y la reusabilidad de componentes. La formalización es una de sus debilidades.
- **Evolución y sostenibilidad:** La evolución del Centro está delineada a grandes rasgos y se realizará por etapas, garantizando que la organización se enfrente a los retos para los que esté preparada. Estas etapas poseen macro-indicadores bien definidos que evidenciarán el momento de cambio a una superior. La etapa actual comienza explotando la diversificación de tecnologías previamente desarrolladas, proceso que sirve de base además a la consolidación de la preparación los recursos humanos y la adquisición y ampliación de su infraestructura para desarrollos ulteriores y para los

procesos de innovación e investigación que deberán involucrar a por cientos mayoritarios del personal en las etapas futuras. Dentro de 2 meses se llega a los 3 años, por lo que se les considera mas que un proyecto, se han firmado mas de 15 contratos, se tienen 2 en análisis en estos momentos y existen alrededor de 6 contratos nuevos. SCADA es más que un proyecto, es una organización que tributa como cliente fundamental a PDVSA.

- **Rentabilizar el conocimiento:** éste se aplica a partir del rol que juegan los desarrolladores, estos están especializados, conocen bastante sus líneas de trabajo, mantienen buena comunicación con los automáticos y se concretan en los resultados. Hay un grupo técnico que garantiza la orientación de las arquitecturas a componentes y a rehusar los ya desarrollados.

- **Recursos necesarios:**

Recursos tecnológicos: Se necesitan resolver problemas de conectividad; es imprescindible cambiar el switch, que posibilite ampliar el ancho de banda interno y externo. Se requiere comunicación telefónica fija y móvil, nacional e internacional, voz sobre IP para, los líderes de proyectos y para los responsables de las líneas funcionales y responsables de los procesos fundamentales de la organización y de alta capacidad de transferencia por Internet. Se cuentan con 90 estaciones de trabajo, teniendo en cuenta que 7 de ellas son propias.

Recursos humanos: Cuenta con 260 desarrolladores, pero solo menos de 100 aportan sustancialmente al trabajo. El personal siempre es bienvenido al proyecto, pues permite especializarse y discernir roles.

- **Personal. Capacitación del personal propio:** Se cuenta con el Dpto. de Sistema Digital que se encarga de la capacitación, adiestramiento y preparación; donde casi todos los profesores del equipo están agrupados en éste. Los trabajadores están agrupados en cinco líneas funcionales y jerarquizadas en cuatro categorías liderado por un grupo de especialistas con más de cinco años en el desarrollo de proyectos informáticos. Los grupos funcionales agrupan los recursos que se especializan en determinadas temáticas que son fundamentales para la organización, y están divididos en Inteligencia del Negocio, Grupo Técnico, Grupo Metodológico, Grupo de Hardware y Grupo de Formación. El centro cuenta con las categorías de técnico, técnico superior, especialista y especialista superior y trabaja con asesores de otras entidades que aportan el conocimiento de los negocios que se automatizan y otras competencias necesarias.

Sobre el entorno (mercado (tendencias), competidores, distribuidores u otros factores, características de los países entorno legal, político, económico)

- **Relación con los Clientes:** se basa en brindar una atención especializada a los clientes en ambientes distribuidos de desarrollo que requieren de la comunicación directa y amplia entre diferentes equipos en locaciones distantes mediante una red privada virtual. No se usa marketing, se está intentando fomentar una estructura que atienda directamente el marketing, esta es una misión que se redefine constantemente de acuerdo con la experiencia de trabajo.
- **Competencia:** Consolidar en sus soluciones el empleo de hardware propio, asociarse a otras empresas y entidades nacionales o extranjeras que puedan aportar mercados y/o el completamiento de las competencias o tecnologías que se necesiten para pasar a nuevos estadios de desarrollo. Los sistemas SCADAs a nivel internacional se basan en ofrecer servicios dirigidos y orientados al diseño, configuración, instalación, puesta en marcha y evaluación de sistemas SCADA para monitoreo y supervisión de sistemas de medición de inventario de volumen y para sistemas de medición de flujo. Entre ellos se encuentra **IntegraXor** es el galardonado programa HMI/SCADA basado en web, **Winlog Lite** es un software de SCADA/HMI poderoso y fácil de usar, para la supervisión de plantas industriales y civiles, así como **Likindoy** es un programa diseñado para análisis industrial y sistemas de telecontrol, así como la gestión de automatizaciones. El proyecto Likindoy pretende crear un Sistema SCADA completo realizado con tecnologías libres: GNU/Linux, Python, MySQL y R.
- **Público objetivo:** Productos y servicios orientados a satisfacer necesidades de:
 - Supervisión y Control en:
Industrias.
 - Petrolera.
 - Energética.
 - Alimenticia.
 - Azucarera.
 - Hospitales.
 - Hoteles.
 - Edificios.

CAPÍTULO 3. TRES PROPUESTAS PRINCIPALES DE MODELOS DE NEGOCIO PARA EL SWL A UTILIZAR EN LA UCI.

- **Formas de entrada al mercado:** fundamentalmente se tiene un cliente externo (Mediante el Convenio Cuba-Venezuela y PDVSA-ALBET) y varios clientes nacionales (organismos y entidades del país). Informática 2009 sirvió para darse a conocer el proyecto, por lo que algunas entidades han comenzado a interesarse.
- **Relaciones con Comunidades y Foros:** está vinculado al Polo Productivo de Hardware y Automatización de la Universidad de las Ciencias Informáticas. Comunidad de SWL y OPC.
- **Experiencia anterior.**
 - **Obtención de beneficios:** Se ingresaron al país en el primer año y por conceptos de investigaciones principalmente 1.2 millones de dólares y se ha ingresado al país actualmente cerca de 2.5 millones de dólares con un gasto no mayor que 200 000 dólares lo que representa haber obtenido más del 92% de ganancias para el país. Los resultados obtenidos hasta ahora y el prestigio que ha ganado el equipo de trabajo en Mérida han asegurado la posición como equipo de desarrollo para la solución contratada y ha ayudado a alcanzar la Soberanía Tecnológica de Venezuela.

ELEMENTOS QUE FUNDAMENTAN EL MODELO DE NEGOCIO

- **Ventajas:**
 - A nivel mundial son muy pocos los sistemas SCADAs que están desarrollados con tecnología libre.
 - Posee una buena estructura organizativa que evoluciona de acuerdo a los nuevos rumbos de la tecnología y el mercado y la investigación se sitúa en el lugar preponderante dentro de ella.
 - Cuentan con una basta experiencia identificando productos de demanda que estén al alcance, con bajo nivel de inversión, acordes a nuestro desarrollo y a las indiscutibles fortalezas en áreas del conocimiento del país (Medicina, Deportes, Educación, Defensa y Seguridad) e incorporarlos a su cartera.
 - Se destacan por la rapidez de respuesta y asumiendo retos en cortos plazos de contrato para satisfacer las necesidades del cliente.

- **Desventajas:** En los servicios de formación tienen su mayor debilidad. Requieren de mayor ancho de banda en cuanto las conexiones de red para estar en constante intercambio con los clientes.

3.4 Conclusiones parciales

En este capítulo se partió del análisis y estudio de los Modelos del capítulo 1, de las fichas del Anexo 1 y de las diferentes maneras de comercialización que existen en la UCI, para lograr fusionar de forma general los modelos que pueden ser aplicados en casos puntuales de la universidad, conformando una serie de fichas ordenadas alfabéticamente, de las que sólo se escogieron tres, para aplicarlas a tres proyectos productivos como UNICORNIOS, FILPACON y SCADA para PDVSA, bajo un mismo formato que indica cómo identificar sus clientes, cómo ofertar sus productos/servicios, de qué forma entra esta cartera de productos al mercado, cuáles son los recursos que necesitan para ofrecer esto y cuál es el beneficio que genera. Esto demuestra que a medida que cambia la competencia, las relaciones con los clientes/usuarios y las tecnologías, también los Modelos de Negocio cambian, según este juego de relaciones. De esta manera, además de traducir la forma de hacer negocios, a la hora de aplicar un modelo, el desarrollador puede conocer la forma en que los negocios van a producirse.

Es necesario que los desarrolladores de SWL en la universidad aprendan ajustar sus Modelos de Negocio según las necesidades de sus clientes, para lograr compartir, colaborar y pensar en las alianzas estratégicas para hacer negocios de una forma más estilizada, abarcadora y ajustada a sus propias necesidades. En este sentido, tanto la participación como la colaboración son vitales, ya que esto depende de la credibilidad de sus desarrolladores a la hora de ofrecer sus servicios o productos en el mercado.

CONCLUSIONES GENERALES

Con el desarrollo de las tres propuestas de Modelos de Negocio para SWL, a partir de la investigación realizada para la aplicación de las mismas a diferentes Casos de Estudio y guiándose por las diferentes maneras que existen para efectuar negocios con el SWL, se arriba a las conclusiones siguientes:

1. La Universidad desarrolla algunos productos bajo licencias libres, éstas no representan costo alguno a la hora de comercializarlos; lo que significa un ahorro para la UCI.
2. Se demuestra que estos países de América Latina tienen cierta dependencia tecnológica que no les permite desarrollarse y por ello el SWL constituye una vía de solución para alcanzar aquellos logros que tanto anhelan.
3. El SWL es rentable y seguro, aunque el desconocimiento por parte de algunos desarrolladores del centro con respecto a temas legales, el mercado al que van dirigidos los productos/servicios y su competencia se hacen evidente.
4. En la UCI los desarrollos de SWL son comercializados a través de la empresa ALBET; que será la encargada de validar los Modelos de Negocio que puedan ser aplicados a algún servicio/producto.
5. Las investigaciones y encuestas que se realizaron sobre la elaboración de un Modelo de Negocio para la comercialización del SWL en la UCI permitieron conocer la falta de experiencia de los desarrolladores de los proyectos productivos que basan sus desarrollos en SWL.
6. Las entrevistas realizadas a profesionales de la UCI y a Empresas Asociadas confirmó la necesidad de elaborar Modelos de Negocio para comercializar SWL.

RECOMENDACIONES

Pueden existir muchas formas de obtener ganancias a través del SWL, que pueden ser agregadas a las usadas hoy en día y que no se explotando. A partir de los resultados o beneficios que proporciona este trabajo de diploma, se proponen las siguientes recomendaciones:

- Crear un Centro de Negocios para SWL en la UCI, donde se brinden servicios de calidad de SWL, se combinen herramientas, plataformas, estudios sobre distintos temas, entre otros aspectos.
- Ofrecer entrenamientos y otorgar certificaciones (como la Certificación LPI), para mejorar la preparación de los desarrolladores de la Universidad y obtener un excelente prestigio internacional.
- Intensificar el trabajo con los estudiantes y profesionales con respecto a los temas legales, es decir, licencias y compatibilidades entre ellas, esto es importante a la hora de comercializar el SWL.
- Que este Trabajo sea tomado como material de consulta por el personal (dígase técnicos o profesionales) que apliquen modelos similares en el futuro.
- Impartir cursos optativos relacionados con el objetivo general del trabajo para profundizar los conocimientos de los estudiantes, miembros de proyectos y desarrolladores.
- Aplicar estos modelos a todos los productos/servicios de SWL que se quieran comercializar para obtener mayores beneficios para la Universidad, así como para el país.
- Convocar un evento con el objetivo de buscar otras vías de efectuar negocios con el SWL, éste puede ser llamado “Concurso de Propuestas Económicas para Comercializar SWL”, y puede ser convocado a través del propio Centro de Negocios para SWL.

BIBLIOGRAFÍA

1. Aveleira Rodríguez, Yanicet. *Contribución a la Epistemología del Software Libre*. [pdf] Habana, Cuba : <http://biblioteca.uci.cu> , 2008. 1.
2. Feller, et al. *Perspectives on Free and Open Source Software. 1er ed.* Massachusetts: The MIT, London : <http://mitpress.mit.edu/books/chapters/0262562278.pdf>, 2005. 4.
3. Barrios Dueñas, Joel. Haciendo negocio con Software Libre. . Disponible en Tomado el . *Alcance Libre*. [En línea] 12 de Junio de 2007. [Citado el: 3 de febrero de 2009 .] <http://www.alcancelibre.org/> . 8.
4. Sus particularidades en Cuba, 2006. . [aut. libro] O. B Soler. *Análisis jurídico del software libre. Sus particularidades en Cuba*. Cuba : s.n., 2006.
5. GNU. . *Diversas licencias y comentarios sobre ellas*. [En línea] 2009. [Citado el: 29 de enero de 2009.] <http://www.gnu.org/license/license-list.es.html> . 13.
6. Apache . *Apache License, Version 2.0. The Apache Software Foundation*. [En línea] enero de 2004. [Citado el: 29 de enero de 2009.] <http://www.apache.org/licenses/LICENSE-2.0.htm> . 5.
7. Ucipedia . *Proyecto UNICORNIOS* . [En línea] 2009. [Citado el: 5 de marzo de 2009.] http://ucipedia.uci.cu/index.php/Proyecto_UNICORNIOS . 29.

8. *La Propiedad Intelectual y la Industria del Software*. Habana, UCI : s.n., 19 de abril del 2009. 1.
9. WIPO. *WIPO Intellectual Property Handbook: Policy, Law and Use. Chapter 1 – Introduction*. [En línea] 2009. <http://www.wipo.int/export/sites/www/about-ip/en/iprm/pdf/ch1.pdf> .
10. Batista Soler, Otto. *La Explotación de los Derechos de Autor en el Software Libre*. Habana, UCI : s.n., 2009. 2.
11. Schadler, Ted. Brodeur. *Open Source rewards outweigh the risks*. [En línea] 2004. http://www.brodeur.de/de/magazin/index_449.htm. 27.
12. Saravia, D. Hipatia. *Diccionario conceptual del conocimiento libre*. [En línea] 2005. [Citado el: 8 de febrero de 2009.] <http://docs.hipatia.info/diccionario> . 26.
13. Gvsoft . *Acerca de Open Source - Código Abierto*. [En línea] 31 de agosto de 2004. [Citado el: 3 de febrero de 2009.] <http://www.gvsoft.com/Open Source - Software Libre./OpenSource.html> . 6.
14. Moore, J. T. S. Inter-Nos 2001. *Revolution OS, (Documental)*. [En línea] 2001. http://inter-nos.uci.cu/Teleclases/Teleclases.asp?id_as=29 [Ave08]. 17.

15. Najaraba. *Qué da más: libre o abierto.* . [En línea] 11 de mayo de 2006. <http://najaraba.blogspot.com/2006/07/impacto-de-las-distribuciones-de.html> . 18.
16. Kobayashi Gutiérrez, Alfred. *El Modelo de Negocio del Software Libre, como eje en la Industria de Software Peruano para competir en el mercado Mundial.* [pdf] Perú : <http://web2.evolutionperu.com/wp-content/uploads/2007/05/mnsl.pdf> , 21 de octubre del 2006. 5.
17. Díaz Indaba, José Ramón. *Nuevos Modelos de Negocio: Software libre.* [pdf] Mexico : s.n., Noviembre 2005. 3.
18. Gelbort, Román H. Café Conf. *Modelos de Negocio con SWL. 6tas Conferencias Abiertas de Software Libre y GNU/Linux.* [En línea] 6 de octubre de 2007. <http://www.cafeconf.org/2007/modules/myconference/viewsspeech.php?sid=8&cid=1> . 9.
19. *El software como servicio. O de como producir programas libres y no morir en el intento.* Baranda. Revista TodoLinux. 25, Octubre 2002. 1.
20. Francesc Miralles, Guillermo Armelini. Rpte. *“Software” de código abierto, de una iniciativa altruista a una fuente de negocios.* Euskotek. Número 22. [En línea] 4º trimestre 2003. http://www.rpte.net/euskotek/numero_22/Euskotek22.pdf . 15.

21. Hecker, Frank. Hecker. *Setting up shop: The business of opensource software*. [En línea] Mayo de 1998. Disponible en <http://www.hecker.org/writings/setting-up-shop.html> . 10.
22. Stallman, Richard M. *Software libre para una sociedad libre*. s.l. : Traficantes de Sueños, 2004. ISBN: 84-933555-1-8.
23. Alainet . *América Latina en Movimiento. El Software Libre como herramienta de inclusión*. [En línea] 15 de febrero de 2007. <http://alainet.org/active/15626&lang=es> . 3.
24. Documentacion.prod.hab.uci.cu . *Guía de Venezuela*. [En línea] 2008. [Citado el: 27 de febrero de 2009.] <http://documentacion.prod.hab.uci.cu/Literatura/Cientifica/Migracion/guiadevenezuela> . 30.
25. Ministerio de Ciencia y Tecnología. *Plan Nacional de Migración al Software Libre de la Administración Pública Nacional Oficina de Tecnologías de Información*. [En línea] Marzo de 2005. [Citado el: 27 de febrero de 2009.] <http://www.mct.gov.ve/uploads/biblio/PLANNACIONALDEMIGRACIONASWL230305.pdf> . 22.
26. *Decreto N° 3.390*. Gaceta Oficial . 38.095, Venezuela : <http://fsfla.org/svnwiki/legis/venezuela/xsvnwiki-atom/3390.es> , 28 de diciembre del 2004 . 3.
27. La flecha. *Noticias*. [En línea] marzo de 2009. [Citado el: 3 de marzo de 2009.] Disponible en <http://www.laflecha.net/canales/softwarelibre/noticias/200503311.htm> . . 18.

28. Master Magazine . *artículo*. [En línea] marzo de 2009. [Citado el: 3 de marzo de 2009.] <http://www.mastermagazine.info/articulo/9234.php.htm> . 6.
29. Mazoni, Marcos. Entrevista con Marcos Mazoni. *Modelos de Negocio con SWL*. 15 de enero de 2009.
30. Zorzoli, Pablo Luis. Colección.edu. *Investigación sobre el Movimiento del Software Libre*. [En línea] http://coleccion.educ.ar/coleccion/CD7/img/docs/acerca_soft/mod01/mod01_softwarelibrepz/indice.html.
31. El siglo de torreon . *Promete crecimiento software libre en Mexico*. [En línea] 2009. [Citado el: 6 de marzo de 2009.] <http://www.elsiglodetorreon.com.mx/noticia/336936.promete-crecimiento-software-libre-en-mexico.html> . 28.
32. Master Magazine. [En línea] marzo de 2009. [Citado el: 6 de marzo de 2009.] <http://www.mastermagazine.info/articulo/11277.php> . 15.
33. *Mini-Guía de Migración a SWL*. Unicornios, Proyecto. Habana, Cuba : Universidad de las Ciencias Informáticas, 20 de mayo 2008. 9.
34. Portal de SWL. *Historia de la UCI*. [En línea] 2009. http://ucipedia.uci.cu/index.php/Historia_de_la_Comunidad_de_Software_Libre_de_la_UCI . 11.

35. López Cosío, MSc. Filiberto. Conferencia en el evento de la Comunidad Cubana de PosgretSQL. “*La producción de Software en la Universidad de las Ciencias Informáticas (UCI)*”. UCI, Habana : Teatro del Docente 3, Viernes 6 de marzo de 2009.
36. Abreu, Marta. *Entrevista a la Ing. Marta Abreu.* . Habana, Cuba, 2 de marzo de 2009.
37. Piñero Pérez, Pedro Yobanis. *Infraestructura Productiva de CENTALAD.* Habana, UCI : Grupo de Gestión de Negocios (CENTALAD), 4 de marzo de 2009. 2.
38. Nova. *Página Principal.* [En línea] 2009. <http://www.nova.uci.cu/website/?q=node/12.20>.
39. Oropesa Méndez, Daisy. Entrevista con Daisy Oropesa. *Formas de comercialización de SWL en la UCI.* Habana, Cuba : Oficina de ALBET, Docente 5, UCI, 12 de marzo de 2009.
40. Meneses, Abel. Entrevista realizada al Ing. Abel Meneses . *Modelos de Negocio con SWL.* Habana : Laboratorio 102, Docente 3 de la UCI., 10 de marzo de 2009.
41. *Segunda Conferencia Internacional de SWL. Modelos de Negocio para Software Libre.* s.l. : <http://merida05.opensourceworldconference.com/es> , agosto, 2005. 1.

42. Monera Daroqui, Fernando. Open Sistemas. *Modelos de Negocio basados en Software Libre*. [En línea] 8 de septiembre de 2003. <http://www.opensistemas.com/> . 16.
43. WordPress . *Tres Modelos de Negocio para el software libre*. [En línea] 19 de noviembre de 2008. <http://tres-modelos-de-negocios-para-el-software-libre.htm> .
44. Rossi, Andrea, Gonzales, Pascual y Iglesias, Carlos. Ines. *Modelos de Negocio para SW libre*. [En línea] febrero de 2007. http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc . 25.
45. Fainelli Florian, et al. *Sistemas empotrados y Hardware abierto de los RMLL*. [En línea] 2009. <http://2009.rml.info/> .2009.
46. Laurière Stéphane, et al. Cualipso. *Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software*. . [En línea] 28 de 04 de 2008. <http://www.qualipso.org/businessmodels> . 12.
47. Thüer, Sebastián. Thuer. *Tres Modelos de Negocio para el software libre*. [En línea] 19 de noviembre de 2008. [Citado el: 29 de enero de 2009.] <http://www.thuer.com.ar/tres-modelos-de-negocios-para-el-software-libre.htm> .

48. Barahona, Jesús, González, Pascual y Seoane, Joaquín. *Introducción al software libre*. [pdf] Madrid : http://www.documentacion.prod.hab.uci.cu/Literatura/Cientifica/Maestria_Internacional_SWL , noviembre, 2003.
49. Alcove. [En línea] febrero de 2009. <http://www.alcove.com> . 4.
50. Abadia Digital. *Así se financian los grandes proyectos del software libre*. [En línea] 13 de Nov de 2008. [Citado el: 10 de enero de 2009.] <http://www.abadiadigital.com/noticia3377.html> . 1.
51. Ximian. . [En línea] 2009. <http://ximian.com/> .
52. Zope corporation. [En línea] 2009. <http://www.zope.com/> .
53. Ada core technologies. [En línea] febrero de 2009. <http://www.gnat.com> . 2.
54. Merit, et al. *Study on the: Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU*. EU : s.n., 2006. 8.
55. Maemo. [En línea] 2008. <http://maemo.org/> . 14.

56. Red Hat. [En línea] 2009. <http://www.redhat.com> . 23.
57. Ortiz Balmaceda, Marcos Luís. Revista de Software libre. *Modelos de Negocio, Proyecto Unicornios (UCI)*. [En línea] 2008. [Citado el: 5 de marzo de 2009.] /etc/init.d/uxi start/ No.08 Vol.02/. 21 .
58. Meneses Abad, Abel. *Migración ¿Qué proyectos usan y desarrollan Software Libre en la UCI?* . Habana, UCI : Revista de SWL de la UCI, Marzo 2007.
59. Producingoss. *Compatibilidad entre licencias*. [En línea] 2009. [Citado el: 5 de abril de 2009.] <http://producingoss.com/es/license-aspects.html> . 7.
60. GNU GENERAL PUBLIC LICENSE. [Citado el: 20 de abril de 2009.] <http://www.gnu.org/licenses/gpl-3.0.html>

Otras bibliografías consultadas

1. Word Press. *El software libre en América Latina*. [En línea] 16 de noviembre de 2007. <http://alaingonza.wordpress.com/2007/11/16/el-software-libre-en-america-latina/> .
2. Coleccion.educ . *Software Libre en Argentina*. [En línea] 2009. http://coleccion.educ.ar/coleccion/CD7/img/docs/acerca_soft/mod01/mod01_softwarelibrepz/indice.html .
3. Armendaris, L. M. Guimi.net . *Sobre el código abierto (open source)*. [En línea] 2006. http://guimi.net/descarga/tec-docs/Sobre_el_OS.pdf .
4. Artofcode LLC. . [En línea] 2009. <http://artofcode.com> .
5. Asay, Matt. Blogs.CNET. *Are proprietary maintenance fees worse than open-source maintenance fees?* [En línea] 2007. http://blogs.cnet.com/8301-13505_1-9827846-16.html .
6. Blanco, Elena. OSS-Watch. *Dual License, University of Oxford*. [En línea] 2006. http://www.oss-watch.ac.uk/resources/duallicence2.xml#body.1_div.2 .
7. Branco, Marcelo. Software Livre. *Software Libre Brasil "Vamos a Tropicalizar la Digitalización" - Versión: 1.0*. [En línea] 30 de Jun de 2003. <http://www.softwarelivre.org/> .
8. Baquia. *El software libre propaga su mensaje desde Brasil*. [En línea] 22 de enero de 2009. <http://www.baquia.com/el-software-libre-propaga-su-mensaje-desde-brasil.htm> .
9. LPI. *Certificación LPI*. [En línea] 2009. [Citado el: 16 de marzo de 2009.] <http://lpi.org.es/certificacionLPI/QuéésLPI.mht> .

10. Brasi, Claudio De. Blogspot. *Modelo de negocios con Software Libre. CafeConf*. [En línea] 2007. [Citado el: 18 de abril de 2009.] <http://ul64.blogspot.com/2007/10/modelo-de-negocios-con-software-libre.html> .
11. Portal Software Libre . *Historia de la Comunidad de Software Libre de la UCI*. [En línea] 2009. <http://softwarelibre.uci.cu> .
12. Teleformación. *Conferencia 2. FT Modelamiento del Negocio*. [En línea] 2007. [http://teleformacion.uci.cu/ Preparaci3n para la Prueba de Nivel de IS/Conferencias.rar](http://teleformacion.uci.cu/Preparaci3n+para+la+Prueba+de+Nivel+de+IS/Conferencias.rar) ..
13. Free Software Foundation . *What is Copyleft?* [En línea] 2009. [Citado el: 29 de enero de 2009.] <http://www.fsf.org/licensing/essays/copyleft.html> .
14. Codigo Sur. *Cuba pide software libre*. [En línea] 8 de septiembre de 2007. [Citado el: 2 de marzo de 2009.] <http://www.codigosur.org/indice.php/> .
15. Cusumano, Michael A. Ebusiness. *The Changing Software Business: Moving from Products to Services*. [En línea] enero de 2008. http://ebusiness.mit.edu/research/papers/236_Cusumano_Changing_Software_Business.pdf .
16. Ucipedia. *CuteOS - Sistema operativo para la Web*. [En línea] 2009. [http://ucipedia.uci.cu/index.php/CuteOS - Sistema operativo para la Web](http://ucipedia.uci.cu/index.php/CuteOS_-_Sistema_operativo_para_la_Web) .
17. Hipatia. *Definiciones y conceptos relativos al Software Libre. Diccionario temático Hypatia*. [En línea] octubre de 2003. <http://www.hipatia.info> .
18. Delio, Michelle. Wired News. *El código abierto no es algo nuevo en la red*. [En línea] Enero de 2003. [Citado el: 5 de diciembre de 2008.] <http://us.terra.wired.com/news/linux/0,1274,21105,00.html> .

19. EMule-Project.net . *Sitio Oficial de eMule. ¿Qué es eMule?* [En línea] 2009. [Citado el: 29 de enero de 2009.] <http://www.emule-project.net/emule/general.cgi.htm> .
20. Eweek. [En línea] 2008. <http://www.eweeek.com/article2/0,1895,1442452,00.asp> .
21. Feltrero, Roberto. Sindominio. *Las licencias del software libre: nuevos modelos y filosofías para la Propiedad Intelectual.pdf*, Madrid. [En línea] Abril de 2003. [Citado el: 10 de enero de 2009.] <http://sindominio.net/copyleft> y http://www.uned.es/dpto_log/rfeltrero .
22. Ferrer, Alejandra. Mosaic.uoc. *Hacia la libertad del conocimiento... ¿Llegaremos?* [En línea] 18 de enero de 2002. [Citado el: 5 de diciembre de 2008.] http://mosaic.uoc.edu/old/2_opinion2/modeljuny/alejandraferrer.html ..
23. GNU.org. *Free Software Definition*. [En línea] <http://www.gnu.org/philosophy/free-sw.html> ..
24. FSFLA. *Objetivos de la Fundación Software Libre América Latina (FSFLA)*. [En línea] 5 de mayo de 2008. <http://www.fsfla.org/svnwiki/about/objectives.es> .
25. Galli, Ricardo. MNM.uib. *Aspectos Éticos del Software Libre.pdf*. [En línea] [Citado el: 5 de diciembre de 2008.] <http://mnm.uib.es/gallir/> .
26. González Barahona, Jesús M., Seoane Pascual, Joaquín y Robles, Gregorio. Sinetgy. *Software libre: licencias y propiedad intelectual.pdf*. [En línea] julio de 2004. <http://sinetgy.org/jgb> .
27. Grau Merconchini, David. Juventud rebelde. *Software libre III: ¿Cómo estamos seguros?* . [En línea] 28 de febrero de 2008. <http://juventudrebelde.cu> .
28. Hardings, J y A., Fuentes. Hardings. *Software Libre: introducción histórica*. [En línea] 2003. <http://www.hardings.cl/publications/hardings2003intro.pdf> .

29. Hermenegildo, Manuel. Clip.dia.fi.upm. *Problemática y Aspectos Técnicos Sobre La Patentabilidad del Software*. [En línea] 23 de enero de 2003. <http://www.clip.dia.fi.upm.es/herme> .
30. Hierro, Juan José y Jariego, Francisco. Morfeo-Project. *Ejemplo de colaboración entre la Universidad y Telefónica: el proyecto Morfeo.pdf*. [En línea] 2008. <http://www.morfeo-project.org> .
31. Teleformacion. *Historia de los sistemas Operativos. . Tema 3*. [En línea] 2009. <http://teleformacion.uci.cu/HI/Tema3/HistoriadelossistemasOperativos.pdf> , UCI .
32. Open Source. org. *Listado de Open Source Initiative*. [En línea] <http://www.opensource.org/licenses> .
33. Martínez, E. M y Murillo, O. J. F. Eveliux. *Principios y filosofía del software libre*. [En línea] 2007. [Citado el: 10 de enero de 2009.] http://eveliux.com/mx/index.php?option=com_content&task=view&id=187&Itemid=26 .
34. Softwarelibre.fox . *Mexico*. [En línea] 2009. [Citado el: 2009 de marzo de 2009.] <http://softwarelibre.fox.presidencia.gob.mx/?q=node/411> .
35. Obtenga linux.org. *Mitos acerca del software libre*. [En línea] 2009. www.obtengalinux.org .
36. My SQL . [En línea] 2009. [Citado el: 3 de febrero de 2009.] <http://www.mysql.com/about/> .
37. Nokia. [En línea] <http://www.nokia.com> .
38. Nova. *¿Quiénes Somos?* . [En línea] 2009. [Citado el: 5 de marzo de 2009.] <http://www.nova.uci.cu> .
39. Perens, Bruce. Perens . *The Emerging Economic Paradigm of Open Source*. [En línea] 2005. <http://perens.com/Articles/Economic.html> .

40. Portal de Producción. [En línea] http://ucipedia.uci.cu/index.php/Portal_de_Producci%C3%B3n .
41. Portal de Software Libre. . [En línea] http://ucipedia.uci.cu/index.php/Portal_de_Software_Libre .
42. Ucipedia. *Proyecto Entrenadores Aduana*. [En línea] http://ucipedia.uci.cu/index.php/Proyecto_Aduana
43. Raymond, Eric S. Documentacion.prod. *The Cathedral & the Bazaar (paperback) Musings on Linux and Open Source by an Accidental Revolutionary*. [En línea] 2001.
<http://www.documentacion.prod.hab.uci.cu> . 0-596-00108-8 .
44. Red Hat. *Políticas de Privacidad*. [En línea] [Citado el: 15 de enero de 2009.]
http://www.redhat.com/legal/privacy_statement.html .
45. *Migración Pág. 6* . UXi, Revista. 01, Habana, Portal de Software Libre UCI : s.n., febrero, 2008, Vol. 02.
46. *Migración “Estudio de Producción Multimedia con GNU/Linux”* . UXi, Revista. 03, Habana, Portal de Software Libre UCI : s.n., ABRIL / 2008, Vol. 02.
47. *Servicios Libres “Observatorio de Tecnologías Libres”* . UXi, Revista. 04, Habana, Portal de Software Libre UCI : s.n., MAYO / 2008, Vol. 02.
48. *Revista UXi. Las políticas de Seguridad en la distribución NOVA. Pág. 20.* . 07, Habana, Portal de Software Libre UCI : 2008, Vol. 02.
49. *Propuesta de estudio webmétrico sobre impacto y uso del Software Libre en la UCI* *pág 26.* . UXi, Revista. 08, Habana, Portal de Software Libre UCI : s.n., 2008, Vol. 02.

50. Sánchez Acosta, Alejandro. Neurowork.net. *Modelos de Negocio de SWL*. [En línea] 1 de mayo de 2007. [Citado el: 5 de diciembre de 2008.] <http://www.neurowork.net/blogs/asanchez/2007/05/01/modelos-de-negocio-de-software-libre/> .
51. —. *WHYFLOSS Conference*. [En línea] 27 de mayo de 2007. [Citado el: 5 de diciembre de 2008.] <http://neurowork.net/blogs/asanchez/2007/05/27/whyfloss-conference/> .
52. Oracle.com. *Siebel E-Commerce 8.1.1*. [En línea] [Citado el: 3 de febrero de 2009.] <http://www.oracle.com/about/siebel-e-commerce-8-1-1.html> .
53. La Flecha. *Software libre, una alternativa de ahorro en México*. [En línea] 22 de febrero de 2007. <http://www.laflecha.net/> .
54. Tecnoconocimiento. *Software Libre Chile. Nace el Instituto Latinoamericano de Tecnología Libre*. [En línea] 2009. [Citado el: 2 de marzo de 2009.] <http://www.tecnocimiento.cl/ILTL> .
55. The LWN.net . *FAQ*. [En línea] [Citado el: 3 de febrero de 2009.] <http://lwn.net/FAQ.lwn.htm> .
56. Open Sistemas. *TYPO3 CMS*. [En línea] http://www.opensistemas.com/Open_Sistemas TYPO3 CMS.htm .
57. Ec. europa. *Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU*. [En línea] 2006. <http://ec.europa.eu/enterpri> .
58. Varsavsky, Martín. Martin Varsavsky.net. *¿De qué vive el Software Libre?* [En línea] <http://spanish.martinvarsavsky.net/tecnologia-e-internet/ade-qua-vive-el-software-libre.html>.

59. Informática Verde. . *Software libre en las empresas privadas*. [En línea] 25 de julio de 2008. [Citado el: 20 de enero de 2009.]

[http://www.informaticaverde.org/wiki/index.php?title=Software libre en las empresas privadas](http://www.informaticaverde.org/wiki/index.php?title=Software_libre_en_las_empresas_privadas).

60. Najaraba. *Vivir del software ¿libre?* [En línea] 18 de octubre de 2006. [Citado el: 23 de enero de 2009.]

[http://najaraba.blogspot.com/software libre.htm](http://najaraba.blogspot.com/software_libre.htm).

61. Martínez, Segoviano. Ucipedia. *Zamira la alternativa libre de los manejadores de referencia*. [En línea]

<http://ucipedia.uci.cu/index.php/Zotero> .

GLOSARIO DE TÉRMINOS Y SIGLAS

Glosario de Términos

Acuerdo de Licencia con los Contribuyentes: es un formulario electrónico que un desarrollador rellena y envía al proyecto.

Bugs: falla, error en un programa, defecto en un sistema, error en la escritura de un programa lo que produce un daño en el sistema

Cajas negras: un dispositivo o sistema en el cual solo su comportamiento visible externo es considerado y no su implementación.

Ciberespacio: espacio virtual, Internet, donde los usuarios navegan por la web para un beneficio en común.

Copyleft: mecanismo legal con el cual se exige que el software derivado que se pretenda redistribuir sea liberado con licencia igual o “similar” a la del original.

Copyright: derecho de copia en español, proviene de la tradición anglosajona y se basa en la idea de que lo más importante no es el autor, sino la obra y los derechos de propiedad que los autores tienen sobre sus creaciones, con los que se puede comerciar sobre la base de los principios económicos.

Dominio público: ninguna restricción se impone a los usuarios del código fuente. El contenido en cuestión no está protegido y no necesita licencia. En la práctica, si un trabajo está en el dominio público, es como si tuviera una licencia de software libre totalmente permisiva y sin copyleft.

E-learning: educación o entrenamiento a través de Internet.

Empresas partners: empresas asociadas.

Franquicias: una empresa franquicia otorga licencias o marcas, relacionadas con un programa libre determinado.

Freeware: programa de libre acceso, programa con derechos de autor que se puede utilizar sin pago alguno.

Fundación para el Software Libre (Free Software Foundation, FSF): Organización creada a partir del esfuerzo de Richard Stallman y otros entusiastas del software libre con el propósito de difundir este movimiento. La Fundación para el Software Libre (FSF) está dedicada a eliminar las restricciones sobre el

copiado, redistribución, entendimiento, y modificación de programas de computadoras. Hacemos esto, promocionando el desarrollo y uso del software libre en todas las áreas de la computación, pero muy particularmente, ayudando a desarrollar el sistema operativo GNU.

GNU: Proyecto iniciado en 1984 por Richard M. Stallman, que tiene como objetivo desarrollar un sistema operativo compatible con UNIX 100% libre y operacional.

Hacker: término de difícil traducción, mitad genio y mitad pirata, entre virtuoso de la computadora y pirata tecnológico.

Hardware: está integrado por los componentes físicos tangibles que integran a los sistemas de computadoras y sus redes que tienen que seguir un proceso industrial- tradicional hasta alcanzar la producción seriada, donde su protección es a través de patentes y secretos industriales. Entre sus componentes se pueden mencionar: la motherboard, el microprocesador, las memorias RAM y el disco duro, entre otros, que de una forma u otra son los que hacen funcionar la computadora.

Hosting: es el negocio de alojamiento, servicio y mantenimiento de uno o más sitios Web.

Licencia de software : es el instrumento legal a través del cual el autor, titular o proveedor (licenciante) establece las condiciones y los requisitos generales bajo los que se otorga el SW al usuario (licenciataro). Las licencias de SWL y de SW propietario se diferencian entre sí en la amplitud y alcance de las libertades que brindan a los usuarios.

Licenciante: es el autor, titular o proveedor que establece las condiciones y los requisitos generales bajo los que se otorga el SW al usuario (licenciataro).

Linux: Se refiere al Kernel (núcleo, es el corazón del sistema operativo) de un sistema operativo libre, compatible con UNIX (POSIX 100 %), iniciado en 1991 por Linus Benedict Torvalds y desarrollado colectivamente por miles de programadores de todo el mundo. Popularmente se utiliza para referirse a todo sistema operativo que use Linux como núcleo.

Núcleo: en informática, el núcleo, también conocido en español con el anglicismo kernel, de raíces germánicas como kern, es la parte fundamental de un sistema operativo. Es el software responsable de facilitar a los distintos programas acceso seguro al hardware de la computadora o en forma más básica, es el encargado de gestionar recursos, a través de servicios de llamada al sistema. Es la parte del sistema operativo responsable de mantener la fluidez de datos entre los discos duros, la memoria, las impresoras, la pantalla de video y todas las partes que se hallan unidas al mismo.

Plugging: dispositivo de enchufe, tarjeta de programación adicional que se conecta al ordenador fácilmente por medio del Internet en archivos HTML

Redistribuir: volver a distribuir un producto.

Repositorios: un área de almacenamiento para datos y se asocian a un tipo de negocio.

Shareware: los archivos binarios, pero no el código fuente subyacente, son hechos disponibles de manera libre, posiblemente para un período de prueba, solamente.

Sistema Operativo: es una plataforma software, constituida por un conjunto de programas que mediante el cual el usuario puede realizar un sin número de tareas, interactuando con el hardware de la computadora donde fue instalado (incluida la red) y las demás componentes del software.

Software: es un programa ejecutable por los ordenadores mediante un proceso de traducción y compilación que desarrollan los programadores, cuyo fin es interactuar con el hardware y realizar tareas específicas para el usuario. El cual se rige por modelos, políticas y códigos éticos a la hora de su comercialización. Por ejemplo: los sistemas operativos, hojas de cálculo, gestores de bases de datos, procesadores de texto, navegadores de Internet, entre otro. Actualmente existe una gran controversia en los diferentes tipos de software que existen, del cual más adelante se va abundar.

Tangibles: materiales o bienes de valor monetario.

UNIX: es un sistema operativo desarrollado desde hace 30 años, manteniendo el mismo diseño y forma de uso, añadiendo diversas mejoras a lo largo de los años, pero manteniéndose sobre la línea de seguridad y alto rendimiento sobre la cual fue diseñado. TCP/IP, el protocolo básico de Internet, fue construido alrededor de UNIX, por lo tanto la integración de los servicios de Internet en un ambiente UNIX es perfecta.

Usabilidad: La usabilidad son técnicas que ayudan a los seres humanos a realizar tareas en entornos gráficos de ordenador.

Siglas

BSD: Berkeley Software Distribution

CL: Copyleft

CLA: Contributor License Agreement o Acuerdo de Licencia con los Contribuyentes

DEC: Corporación de Equipamiento

DECUS: Grupo de Usuarios de Computadoras DEC

GPL: Licencia Pública General

FEU: Federación Estudiantil Universitaria de Cuba

FSF: Free Software Foundation o Fundación de SWL

MIT: Instituto de Tecnología Massachusetts o Massachusetts Institute of Technology

OSI: Open Source Initiative

OSS: Open Source Software

PC: Personal Computer o Computadora Personal

SCA: Software de Código Abierto

SO: Sistema Operativo

STI: Solución Tecnológica Integral

SW: Software

SWL: Software Libre

TIC: Tecnologías de la Información y las Comunicaciones

UCI: Universidad de las Ciencias Informáticas

UML: Unified Modeling Language o Lenguaje de Modelado Unificado

USD: Dólar de los Estados Unidos

APÉNDICES

Apéndice 1: Decreto No.3.390 – Venezuela*

336.626 GACETA OFICIAL DE LA REPUBLICA BOLIVARIANA DE VENEZUELA Martes 28 de diciembre de 2004

Refrendado
El Ministro de Finanzas
(L.S.) NELSON JOSE MERENTES DIAZ

Refrendado
El Ministro de la Defensa
(L.S.) JORGE LUIS GARCIA CARNEIRO

Refrendado
La Encargada del Ministerio
de la Producción y el Comercio
(L.S.) EDMEE BETANCOURT DE GARCIA

Refrendado
El Ministro de Educación Superior
(L.S.) SAMUEL MONCADA ACOSTA

Refrendado
El Ministro de Educación y Deportes
(L.S.) ARISTOBULO ISTURIZ ALMEIDA

Refrendado
El Ministro de Salud y Desarrollo Social
(L.S.) FRANCISCO ARMADA

Refrendado
La Ministra del Trabajo
(L.S.) MARIA CRISTINA IGLESIAS

Refrendado
El Ministro de Infraestructura
(L.S.) RAMON ALONZO CARRIZALEZ RENGIFO

Refrendado
El Encargado del Ministerio
de Energía y Minas
(L.S.) ORLANDO ORTEGANO QUEVEDO

Refrendado
La Ministra del Ambiente
y de los Recursos Naturales
(L.S.) ANA ELISA OSORIO GRANADO

Refrendado
El Encargado del Ministerio
de Planificación y Desarrollo
(L.S.) JOSE FELIX RIVAS ALVARADO

Refrendado
La Ministra de Ciencia y Tecnología
(L.S.) MARLENE YADIRA CORDOVA

Refrendado
El Ministro de Estado para la
Coordinación y Control de las
Zonas Especiales de
Desarrollo Sustentable
(L.S.) JOSE FRANCISCO NATERA MARTINEZ

Refrendado
El Ministro de Estado
para la Cultura
(L.S.) FRANCISCO DE ASIS SESTO NOVAS

Decreto N° 3.390 23 de diciembre de 2004

HUGO CHAVEZ FRIAS
Presidente de la República

De conformidad con lo dispuesto en los artículos 110 y 226 de la Constitución de la República Bolivariana de Venezuela, 12 y 47 de la Ley Orgánica de la Administración Pública y, 2°, 19 y 22 del Decreto con Rango y Fuerza de Ley Orgánica de Ciencia, Tecnología e Innovación, en Consejo de Ministros,

CONSIDERANDO

Que es prioridad del Estado incentivar y fomentar la producción de bienes y servicios para satisfacer las necesidades de la población,

CONSIDERANDO

Que el uso del Software Libre desarrollado con Estándares Abiertos fortalecerá la industria del software nacional, aumentando y fortaleciendo sus capacidades,

CONSIDERANDO

Que la reducción de la brecha social y tecnológica en el menor tiempo y costo posibles, con calidad de servicio, se facilita con el uso de Software Libre desarrollado con Estándares Abiertos,

CONSIDERANDO

Que la adopción del Software Libre desarrollado con Estándares Abiertos en la Administración Pública y en los servicios públicos, facilitará la interoperabilidad de los sistemas de información del Estado, contribuyendo a dar respuestas rápidas y oportunas a los ciudadanos, mejorando la gobernabilidad,

CONSIDERANDO

Que el Software Libre desarrollado con Estándares Abiertos, permite mayor participación de los usuarios en el mantenimiento de los niveles de seguridad e interoperatividad.

DECRETA

Artículo 1°. 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.

Artículo 2°. A los efectos del presente Decreto se entenderá por:

Software Libre: Programa de computación cuya licencia garantiza al usuario acceso al código fuente del programa y lo

* Tomado el 28 de febrero del 2009 de [45].

autoriza a ejecutarlo con cualquier propósito, modificarlo y redistribuir tanto el programa original como sus modificaciones en las mismas condiciones de licenciamiento acordadas al programa original, sin tener que pagar regalías a los desarrolladores previos.

Estándares Abiertos: Especificaciones técnicas, publicadas y controladas por alguna organización que se encarga de su desarrollo, las cuales han sido aceptadas por la industria, estando a disposición de cualquier usuario para ser implementadas en un software libre o propietario, promoviendo la competitividad, interoperatividad o flexibilidad.

Software Propietario: Programa de computación cuya licencia establece restricciones de uso, redistribución o modificación por parte de los usuarios, o requiere de autorización expresa del Licenciadador.

Distribución Software Libre desarrollado con Estándares Abiertos para el Estado Venezolano: Un paquete de programas y aplicaciones de Informática elaborado utilizando Software Libre con Estándares Abiertos para ser utilizados y distribuidos entre distintos usuarios.

Artículo 3°. En los casos que no se puedan desarrollar o adquirir aplicaciones en Software Libre bajo Estándares Abiertos, los órganos y entes de la Administración Pública Nacional deberán solicitar ante el Ministerio de Ciencia y Tecnología autorización para adoptar otro tipo de soluciones bajo las normas y criterios establecidos por ese Ministerio.

Artículo 4° El Ministerio de Ciencia y Tecnología, adelantará los programas de capacitación de los funcionarios públicos, en el uso del Software Libre desarrollado con Estándares Abiertos, haciendo especial énfasis en los responsables de las áreas de tecnologías de información y comunicación, para lo cual establecerá con los demás órganos y entes de la Administración Pública Nacional los mecanismos que se requieran.

Artículo 5°. El Ejecutivo Nacional fomentará la investigación y desarrollo de software bajo modelo Software Libre desarrollado con Estándares Abiertos, procurando incentivos especiales para desarrolladores.

Artículo 6°. El Ejecutivo Nacional fortalecerá el desarrollo de la industria nacional del software, mediante el establecimiento de una red de formación, de servicios especializados en Software Libre desarrollado con Estándares Abiertos y desarrolladores.

Artículo 7°. El Ministerio de Ciencia y Tecnología será responsable de proveer la Distribución Software Libre desarrollado con Estándares Abiertos para el Estado venezolano, para lo cual implementará los mecanismos que se requieran.

Artículo 8°. El Ejecutivo Nacional promoverá el uso generalizado del Software Libre desarrollado con Estándares Abiertos en la sociedad, para lo cual desarrollará mecanismos orientados a capacitar e instruir a los usuarios en la utilización Software Libre desarrollado con Estándares Abiertos.

Artículo 9°. El Ejecutivo Nacional promoverá la cooperación internacional en materia de Software Libre desarrollado con Estándares Abiertos, con especial énfasis en la cooperación regional a través del MERCOSUR, CAN, CARICOM y la cooperación SUR-SUR.

Artículo 10. El Ministerio de Educación y Deportes, en coordinación con el Ministerio de Ciencia y Tecnología, establecerá las políticas para incluir el Software Libre desarrollado con Estándares Abiertos, en los programas de educación básica y diversificada.

Artículo 11. En un plazo no mayor de noventa (90) días continuos, contados a partir de la publicación del presente Decreto en la Gaceta Oficial de la República Bolivariana de Venezuela, el Ministerio de Ciencia y Tecnología deberá presentar ante la Presidencia de la República, los planes y programas que servirán de plataforma para la ejecución progresiva del presente Decreto.

Artículo 12. Cada Ministro en coordinación con la Ministra de Ciencia y Tecnología, en un plazo no mayor de noventa (90) días continuos, contados a partir de la aprobación por parte de la Presidencia de la República de los planes y programas referidos en el artículo anterior, publicará en la Gaceta Oficial de la República Bolivariana de Venezuela su respectivo plan de implantación progresiva del Software Libre desarrollado con Estándares Abiertos, acogiéndose a los lineamientos contenidos en aquellos, incluyendo estudios de financiamiento e incentivos fiscales a quienes desarrollen Software Libre con Estándares Abiertos destinados a la aplicación de los objetivos previstos en el presente Decreto. Igualmente, las máximas autoridades de sus entes adscritos, publicarán a través del Ministerio de Adscripción sus respectivos planes.

Los planes de implantación progresiva del Software Libre desarrollado con Estándares Abiertos de los distintos órganos y entes de la Administración Pública Nacional, deberán ejecutarse en un plazo no mayor de veinticuatro (24) meses dependiendo de las características propias de sus sistemas de información. Los Ministros mediante Resolución y las máximas autoridades de los entes que le estén adscritos a través de sus respectivos actos, determinarán las fases de ejecución del referido Plan, así como las razones de índole técnico que imposibiliten la implantación progresiva del Software Libre en los casos excepcionales, de acuerdo a lo establecido en el artículo 3° del presente Decreto.

Artículo 13. El Ministerio de Ciencia y Tecnología establecerá dentro de los planes y programas contemplados en el presente Decreto, mecanismos que preserven la identidad y necesidades culturales del país, incluyendo a sus grupos indígenas, para lo cual procurará que los sistemas operativos y aplicaciones que se desarrollen se adecuen a su cultura.

Artículo 14. Todos los Ministros quedan encargados de la ejecución del presente Decreto, bajo la coordinación de la Ministra de Ciencia y Tecnología.

Dado en Caracas a los veintitrés días del mes de diciembre de dos mil cuatro. Años 194° de la Independencia y 145° de la Federación.

Ejecútese,
(L.S.)

HUGO CHAVEZ FRIAS

Refrendado
El Vicepresidente Ejecutivo
(L.S.)

JOSE VICENTE RANGEL

Refrendado
El Ministro del Interior y Justicia
(L.S.)

JESSE CHACON ESCAMILLO

Apéndice 2: Los cinco Modelos de Negocio de Código Abierto *

the **InformationWeek** **B L O G**
DEFINING THE BUSINESS VALUE OF TECHNOLOGY

NEWS **BLOGS** SOFTWARE SECURITY HARDWARE MOBILITY WINDOWS INTERNET

Blogs: [Mobile Digital Life](#) [Google Outsourcing Security](#) [Microsoft/Windows Apple Unvarnished](#)
[David Berlind's Tech Radar](#) [Virtualization Information Management Storage Content Management](#)
[Green Computing](#) [Fritz Nelson's Instigator](#) [Analytics](#) [Government IT](#) [Unified Communications](#) [Clo](#)

 **STARTUP CITY**
By John Foley **Emerging And Innovative Companies To Watch**

Topics: [Open Source](#) : [Startup City](#)

 E-mail this page |  Print this page |  BOOKMARK |  Take Us With You |  Buzz up!

The Five Open Source Business Models

Posted by [Andy Dorman](#), Jan 2, 2008 02:03 AM



Open source has become standard in Silicon Valley, with nearly every software startup planning to release at least some code. So far, they've found five main business models:

1. Sell support services. This is the traditional Linux model, prototyped by Red Hat (NYSE: [RHT](#)). It's still a part of most open-source business plans, but on its own it's rarely enough for startups trying to grow. The problem (for the startups) is that anyone can redistribute the code and sell support or consultancy services, so there's nothing to stop an Oracle (NSDQ: [ORCL](#)), an IBM (NYSE: [IBM](#)), or a Novell (NSDQ: [NOVL](#)) from grabbing most of the services revenue.

2. Build (or run) hardware. Free software ought to make hardware more profitable, but relatively few open source companies have taken the hardware route. (Lots of hardware vendors use open-source in their products, of course, but they're not really open-source companies.) The main reason is that installing software on commodity components has even fewer barriers to entry than selling support, as VA Linux showed during the first bubble. Still, some startups have resurrected the idea, notably [Vyatta](#) (router) and [SocialText](#) (wiki appliance).

3. Proprietary components. Many startups now combine proprietary and open-source code, essentially holding back some functionality from what they release for free. The most successful to use this model so far was VMWare competitor [XenSource](#) (now part of Citrix (NSDQ: [CTXS](#))), which gave away the Xen hypervisor but sold its proprietary management software.

* Dorman, Andy. "The Five Open Source Business Models" The Information Week.

http://www.informationweek.com/blog/main/archives/2008/01/the_five_open_s.html [50]

Competitor [VirtualIron](#) does exactly the same thing, collaborating with Citrix on Xen but competing on management. XenSource's success has made this a popular strategy for other open source startups such as [MuleSource](#) (SOA) and [Hyperic](#) (systems management.) It also gives established software vendors a clear path to open source.

4. Dual licensing. Some customers just don't want to follow open-source licenses (usually the GPL), so many open-source vendors will happily sell them proprietary licenses for the same software. This works well for companies like Trolltech and MySQL, and it could become more popular thanks to new open-source licenses that place tighter restrictions on what other vendors can do for free.

For example, the limits on home DRM in GPL v3 are intended to make consumer electronics more open, but they could eventually give open-source companies a revenue stream from DRM vendors who want the code without the license. The (so far little-used) [AGPL](#) could have an even bigger impact, thanks to its requirement that [SaaS users](#) be able to download the source code. The big disadvantage for startups taking this approach is that they can't easily leverage community development, as they need to hold copyright on all code.

5. Advertising. The Mozilla Foundation discovered this almost by accident, when Google (NASDAQ: [GOOG](#)) paid the [Firefox](#) developers so much in referral fees that they had to incorporate as a for-profit. It's also used (along with the other four) by [Digium](#), the main backer of the free [Asterisk](#) PBX, which comes pre-configured to connect to particular IP telephony services. I expect we'll see more startups embrace this idea as SaaS becomes more common and ASPs offer big bucks for customer leads.

Apéndice 3: Licencias GPL v.3 y BSD

LICENCIA BSD

Copyright ©1991, 1992, 1993, 1994 Regentes de la Universidad de California. Todos los Derechos reservados.

Se permite su redistribución tanto en forma de código fuente como en forma binaria, con o sin modificaciones, con tal de que se cumplan las siguientes condiciones:

- 1.Redistribuciones del código fuente deben retener el aviso de copyright declarada arriba, esta lista de condiciones y la denegación de garantía que sigue.
2. Redistribuciones en forma binaria deben reproducir el aviso de copyright declarada arriba, esta lista de condiciones y la denegación de garantía que sigue en la documentación y/u otros materiales que acompañen la distribución.
3. Todo material de promoción que mencione características o el uso de este software debe mostrar el siguiente reconocimiento: Este producto incluye software desarrollado por la Universidad de California, Berkeley y sus contribuidores.
4. No se permite ni el uso del nombre de la Universidad ni el uso de los nombres de sus contribuidores para apoyar o promover productos derivados de este software sin previo permiso específico por escrito.

Los regentes y contribuidores proveen este software ``tal y como está", y deniega cualquier garantía, ya sea expresa o implícita, incluyendo sin limitación las garantías implícitas de comerciabilidad y aptitud para un propósito específico. ni los regentes ni sus contribuidores serán en ningún caso responsables por perjuicios, ya sean directos, indirectos, incidentales, especiales, punitivos o consecuentes (incluyendo sin

limitación la adquisición de bienes o servicios de sustitución; pérdida de uso, datos o ganancias; o interrupción de negocios), sea cual sea su causa y bajo cualquier teoría de responsabilidad, ya sea en contrato, responsabilidad estricta o entuerto (incluyendo de negligencia o de cualquier otra manera) que surja de cualquier manera del uso de este software, aún en el caso de haber sido avisado de la posibilidad de tal perjuicio.

GNU GENERAL PUBLIC LICENSE [60]

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works. The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too. When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

“This License” refers to version 3 of the GNU General Public License.

“Copyright” also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

“The Program” refers to any copyrightable work licensed under this License. Each licensee is addressed as “you”. “Licensees” and “recipients” may be individuals or organizations.

To “modify” a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a “modified version” of the earlier work or a work “based on” the earlier work.

A “covered work” means either the unmodified Program or a work based on the Program.

To “propagate” a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To “convey” a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as

your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or

warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to “keep intact all notices”.
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an “aggregate” if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
- b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
- c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and non-commercially, and only if you received the object code with such an offer, in accord with subsection 6b.
- d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
- e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A “User Product” is either (1) a “consumer product”, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for

incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, “normally used” refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

“Installation Information” for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

“Additional permissions” are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered “further restrictions” within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have

received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's “contributor version”.

A contributor's “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by

procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is “discriminatory” if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License “or any later version” applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF

MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each

source file to most effectively state the exclusion of warranty; and each file should have at least the “copyright” line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>

This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands ``show w'` and ``show c'` should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

ANEXOS

Anexo 1: Fichas de Modelos de Negocio

Modelo 1: Comercialización con licencias duales.

Ficha 1.1:

- **Modelo:** Modelo de desarrollo con doble licencia.
- **Autor, Fuente, Fecha:**
 1. XOOPS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 [41]
 2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 [42]
- **Descripción:** Básicamente se trata de asociar dos licencias a un determinado código, una de ellas es libre y la otra propietaria. Como usuario, puedes elegir cualquiera de las dos licencias para aplicar al uso del software.
- **Aplicado en:** Open Office
 - **Trolltech:** Estas se distribuyen bajo licencia GPL para UNIX siempre que se desarrolle software libre con ellas. También, puede entregar las mismas librerías bajo licencia propietaria para el desarrollo de software propietario, por lo que si una empresa no desea liberar el código que desarrolló no está obligada a asignar la licencia GPL a sus desarrollos, sino que puede optar por obtener una licencia de Trolltech que le permita desarrollar sus aplicaciones sin heredar la licencia GPL.
- **Ventajas:** La ventaja de pagar una licencia por un producto que se adquiere con licencia gratuita es que se puede utilizar el producto para integrarlo dentro de un software propietario y luego distribuirlo con licencia propietaria.
- **Desventajas:** si se adquiere con licencia de código propietario se limitan los privilegios para re-licenciar el producto.
- **Valoración económica:** -
- **Comentarios:** Este modelo se ha comprobado que es exitoso porque las licencias duales estimulan la introducción del software de código abierto en los negocios mientras impulsa su desarrollo con los ingresos que brinda la venta de licencias propietarias.

Ficha 1.2:

- **Modelo:** Nuevos productos derivados de licencias tipo BSD
- **Autor, Fuente, Fecha:**

1. XOOFS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 **[idem, 41]**
2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 **[idem, 42]**
 - **Descripción:** consiste en que los productos derivados de las licencias tipo BSD pueden re-licenciar el código derivado a cualquier tipo de licencia, incluidas licencias propietarias.
 - **Aplicado en:**
 - MacOS X:** tiene como base el núcleo UNIX de un sistema operativo bajo licencia BSD. Esto ha permitido a Apple la posibilidad de adaptación de este núcleo sin estar forzado a liberar el código fuente.
 - **Ventajas:** con las licencias tipo BSD es posible re-licenciar el código derivado a cualquier tipo de licencia, incluidas licencias propietarias.
 - **Desventajas:** todo el software derivado de la licencia GPL está obligado a llevar la misma licencia.
 - **Valoración económica:-**
 - **Comentarios:** Con este modelo los productos pueden derivarse de la licencia GPL o de BSD aunque BSD permite re-licenciar el código.

Modelo 2: Comercialización de servicios asociados a un producto.

Ficha 2.1:

- **Modelo:** Soporte y productos alternativos
- **Autor, Fuente, Fecha:**
 1. XOOFS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 **[idem, 41]**
 2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 **[idem, 42]**
- **Descripción:** Este Modelo de Negocio consiste en desarrollar una determinada aplicación bajo licencia libre, ofreciendo personalizaciones y/o servicios específicos sobre ese software.
- **Aplicado en:** Red Hat. A través de un servicio de asistencia técnica personalizada brindan soporte a empresas para sus distribuciones de Linux.
 - **Sendmail:** Modelo de Negocio orientado a la consultoría. Ofrece a sus clientes desarrollos a medida relacionados con Sendmail, parametrizaciones personalizadas y soporte en general.
- **Ventajas:** este modelo resuelve las necesidades específicas del cliente con garantía.

- **Desventajas:** puede ocasionar dependencia de la empresa proveedora.
- **Valoración económica:-**
- **Comentarios:** Los desarrolladores están interesados en implementar nuevas funcionalidades por lo se logra una estrecha relación entre la comunidad de software libre y la empresa, en la que ambas salen beneficiadas.

Ficha 2.2

- **Modelo:** Comercialización de servicios asociados
- **Autor, Fuente, Fecha:**
[WordPress. Tres Modelos de Negocio para el software libre](#) tres-modelos-de-negocios-para-el-software-libre.htm. 19.11.2008 [43]
- **Descripción:** La comercialización de servicios es una alternativa viable cuando se ha generado la necesidad de usar un software o servicio.
- **Aplicado en:** Red Hat soporte para Fedora Linux y Askimet
- **Ventajas:** Una ventaja es que este sistema está al alcance de emprendimientos pequeños y grandes: en cada caso hay que analizar la estructura de costos, el nivel de demanda y los recursos disponibles para satisfacerla.
- **Desventajas:** Los ingresos son a mediano plazo por las inversiones iniciales que se deben asumir.
- **Valoración económica:-**
- **Comentarios:** A pesar de que se debe invertir al principio y de que los ingresos no son inmediatos, tiene grandes beneficios principalmente de los que pagan por un servicio más profesional.

Ficha 2.3

- **Modelo:** Formación
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc. [44]
 2. Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [45]
- 1. **Descripción:** La formación puede estar ofertada con la implantación del software, cuando se actualiza y se introducen nuevas funcionalidades o simplemente para aumentar habilidades de los usuarios.
- **Aplicado en:** Red Hat
- **Ventajas:** gran valor añadido ya que habilita a individuales y empresas a acceder a la prestación otros tipos de servicios.

- **Desventajas:** la imagen de marca sigue perteneciendo a la empresa o entidad legal que apoya a la comunidad de desarrollo del software
- **Valoración económica:** -
- **Comentarios:** este servicio se puede llevar a cabo con los tradicionales seminarios o a través de soluciones basadas en Web (e-learning). Es muy importante en la formación intelectual de los profesionales.

Ficha 2.4

- **Modelo:** Personalización
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-ontent/uploads/2008/04/d62_11.doc. [idem, 44]
 2. Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [46]
- **Descripción:** La personalización es proporcionada por las compañías expertas en el proceso de compilar e integrar productos de código abierto implantados en el sistema de destino al cual cobran por el servicio.
- **Aplicado en:** Yaco
- **Ventajas:** la solución de código abierto es más flexible para este tipo de servicio que el uso de software propietario.
- **Desventajas:** el coste derivado de dicha personalización es muy inferior al de desarrollar desde el inicio una herramienta.
- **Valoración económica:** -
- **Comentarios:** El desarrollo y la personalización puede combinarse con formación y soporte sobre las aplicaciones instaladas. Esto se hace para cubrir necesidades específicas

Ficha 2.5

- **Modelo:** Entrenamiento y certificación
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-ontent/uploads/2008/04/d62_11.doc. [idem, 44]
 2. Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [idem, 46]
- **Descripción:** El entrenamiento y las ofertas de certificación preparan a los usuarios y futuros administradores del software para el uso del mismo.

- **Aplicado en:** QINDEL GROUP en España
- **Ventajas:** entrenamiento para expandir las habilidades
- **Desventajas:** Necesitan de personal altamente calificado para ofrecerlo, sino lo tienen deben invertir en tiempo y dinero en la formación de estos antes de brindar este servicio.
- **Valoración económica:** -
- **Comentarios:** Esto es importante para aquellas empresas que desean comercializar su SWL a través de los servicios, donde sus profesionales deben estar bien preparados y certificados, para que el cliente se sienta seguro a la hora de comprarlo.

Ficha 2.6

- **Modelo:** Control de riesgo y seguridad
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc. [idem, 44]
 2. Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [idem, 46]
- **Descripción:** Las compañías se especializan en la valoración de riesgo, en asegurar a las compañías usando OSS (Open Source Software) o en los sistemas que ellos vendan, o como usuarios finales. Tales compañías están listas para encontrar los problemas de fuente abierta antes de que ellos se vuelvan problemas.
- **Aplicado en:**
 - **Open Source Risk Management:** apoyara la fuerza continuada y crecimiento de software de fuente libre y abierta a través de una oferta comprensiva de sofisticados servicios de dirección de riesgo.
- **Ventajas:** proporciona las maneras técnicas para acercarse a posibles fraudes legales asociadas con el uso de SWL.
- **Desventajas:** Los desarrolladores necesitan recursos tecnológicos sofisticados para detectar estos riesgos y mantener actualizado el sistema para evitar posibles puertas traseras.
- **Valoración económica:** -
- **Comentarios:** Con este modelo se pueden identificar y moderar los riesgos que el código pueda tener, para brindarle una mayor seguridad y calidad al software libre que se quiera comercializar.

Ficha 2.7

- **Modelo:** Mantenimiento
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-ontent/uploads/2008/04/d62_11.doc. [idem, 44]
 2. Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [idem, 46]
- **Descripción:** El mantenimiento de una aplicación consiste en mantener el software instalado en un estado consistente y funcionando. También, consiste en instalar y mantener nuevas versiones del producto así como solucionar problemas, entre otros.
- **Aplicado en:** Red Hat
- **Ventajas:** puede ser llevado a cabo dentro de la propia compañía o a través de un proveedor de servicios externo.
- **Desventajas:** se necesita de una estrecha relación con el cliente para responder rápidamente cuando este necesite el servicio por lo que se necesita un mayor ancho de banda para las conexiones de la red o habilitar una página Web que preste servicios las 24 horas, donde este pueda hacer su solicitud.
- **Valoración económica:** -

Comentarios: No existe diferencia entre software propietario y el de código abierto en esta materia, aunque los productos comerciales de software libre ofrecen conjuntamente mantenimiento, soporte y actualizaciones sin ataduras de contratos de mantenimiento.

Ficha 2.8

- **Modelo:** Mejor conocimiento o servicios asociados con ese conocimiento
- **Autor, Fuente, Fecha:**
 1. Thüer, Sebastián. Tres Modelos de Negocio para el software libre Disponible en <http://www.thuer.com.ar/tres-modelos-de-negocios-para-el-software-libre.htm> 19.11.2008 [47]
 2. Barahona, Jesús González; Pascual, Joaquín Seoane y Robles, Gregorio. Introducción al software libre. Noviembre 2003. Disponible en http://www.documentacion.prod.hab.uci.cu/Literatura/Cientifica/Maestria_Internacional_SWL. [48]
- **Descripción:** La empresa que utiliza este Modelo de Negocio trata de beneficiar su conocimiento de un producto (o conjunto de productos) libres. Los servicios que brindan consisten en desarrollos a la medida, adaptaciones o integraciones de los productos en los que son expertas, o bien servicios de consultoría, donde aconsejan a sus clientes cómo utilizar mejor el producto.

- **Aplicado en:**

Alcove: proporciona fundamentalmente servicios de consultoría, consultoría estratégica, soporte y desarrollo para software libre. [49]

- **Ventajas:** La principal ventaja de la empresa está relacionada con el mejor conocimiento del producto que el que tienen sus competidores; por ello, la empresa puede estar bien situada si es productora del producto o participa en el proyecto que lo produce.

- **Desventajas:** El conocimiento por parte del que ofrece el servicio debe de ser especializado en este conocimiento.

- **Valoración económica:** -

Comentarios: Sus ingresos provienen de clientes a los que venden servicios relacionados con ese conocimiento, especialmente desarrollos basados en el producto, modificaciones, adaptaciones, instalaciones e integraciones con otros.

Modelo 3: Micro-pagos por suscripciones.

Ficha 3.1:

- **Modelo:** Suscripciones

- **Autor, Fuente, Fecha:**

1. XOOFS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 [idem, 41]
2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 [idem, 42]
3. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2. Disponible en: http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc 27-02-2007 [idem, 44]

- **Descripción:** Este Modelo de Negocio consiste en que el cliente que utiliza el software libre paga por un conjunto de servicios durante un tiempo determinado, a modo de suscripción. Los servicios más comunes son: el acceso a las actualizaciones, soporte con un tiempo determinado de respuesta, consultoría y formación.

- **Aplicado en:**

Linux Weekly News (lwn.net) : establecieron un sistema de generación de ingresos basado en suscripciones de los usuarios de forma que estos accedieran al boletín semanal y a determinadas noticias con una semana de antelación respecto al resto de los lectores. Tras una semana, estos artículos aparecen disponibles para todo el mundo. [idem, 42]

Alfresco es una herramienta de código abierto para la gestión de contenido empresarial (ECM). El Modelo de Negocio de la empresa se basa en ofrecer la herramienta junto con suscripciones anuales al soporte de la misma. Tiene dos tipos de suscripciones, las Platinum y las Gold. Los tiempos de respuesta son menores y los niveles de servicio más altos en las suscripciones Platinum. **[idem, 44]**

- **Ventajas:** puede ser modificado en función de las necesidades propias de cada proyecto. El pago de suscripciones por parte de los socios tiene ventajas sobre los usuarios normales, como la descarga de software adaptado a distribuciones estándar.
- **Desventajas:** Este Modelo de Negocio está lejos de ser maduro y necesita de un gran número de usuarios para que funcione correctamente.
- **Valoración económica:-**
- **Comentarios:** Con este modelo muchas empresas pueden hacer sus negocios mediante las suscripciones, pagando una tasa anual, donde se le permite tener acceso a información privilegiada, certificación, entre otros.

Modelo 4: Pagos por donaciones.

Ficha 4.1:

- **Modelo:** Los micro-pagos.
- **Autor, Fuente, Fecha:**
- [WordPress. Tres Modelos de Negocio para el software libre](#) tres-modelos-de-negocios-para-el-software-libre.htm. 19.11.2008 **[idem, 43]**
- Así se financian los grandes proyectos del software libre. 13 Nov, 2008 Disponible en <http://www.abadiadigital.com/noticia3377.html>
- **Descripción:** Son emprendimientos personales de tiempo parcial, donde el desarrollador acepta un pago por el tiempo que invierte en el desarrollo de una aplicación.
- **Aplicado en:**
 - **eMule:** contiene un apartado para las donaciones... y no permite efectuar pequeños pagos para sostener el proyecto. También, vende merchandising on-line. **[50]**
 - **Gimp:** no cuenta con una estructura profesionalizada que se encargue de buscar vías para la obtención de ingresos más allá del típico apartado en su Web destinado a captar pequeñas aportaciones por parte de seguidores de este software. **[idem, 50]**
- **Wikipedia Foundation** se financia con muchas donaciones particulares pero también recibe aportes importantes de empresas privadas. Su nivel de dependencia respecto a una fuente es bastante menor.

- **Ventajas:** Pueden recibir importantes aportes de empresas privadas.
- **Desventajas:** El modelo de pagos por donaciones es más inestable y sostener el desarrollo de un emprendimiento de cierta envergadura sobre esta base puede ser demasiado arriesgado.
- **Valoración económica:** -
- **Comentarios:** Es un modelo arriesgado ya que su fuente de ingreso es a través de las donaciones de usuarios o grandes empresas pero si llegara el momento que no se realizara más donaciones perderían la única fuente de ingreso.

Ficha 4.2:

- **Modelo:** El mecenazgo o patrocinio.
 - **Autor, Fuente, Fecha:**
2. [WordPress. Tres Modelos de Negocio para el software libre](#) tres-modelos-de-negocios-para-el-software-libre.htm. 19.11.2008 [**idem, 43**]
 3. Así se financian los grandes proyectos del software libre. 13 Nov, 2008 Disponible en <http://www.abadiadigital.com/noticia3377.html>
- **Descripción:** los grandes proyectos de software libre gestionados por fundaciones tienen su principal fuente de ingresos en los aportes voluntarios de grandes empresas.
 - **Aplicado en:** Firefox y Apache, recurren a corporaciones como Google, Microsoft y Yahoo para recaudar dinero. **Apache:** Es un servidor Web al que se le ofrecen 3 niveles de colaboración en función de la cuantía que estén dispuestas a aportar: Platinum Sponsorship, Gold Sponsorship, Silver Sponsorship, Microsoft, Google y Yahoo donan y forman parte del nivel de patrocinio más alto. [**idem, 50**]
 - **Ventajas:** Este modelo garantiza una regularidad en el flujo de dinero y permite tener un horizonte más despajado ya que la mayoría de los acuerdos se concretan por un cierto número de años.
 - **Desventajas:** Existe la posibilidad de que un proyecto termine por depender de la buena voluntad de una compañía que puede cancelar la financiación cuando le parezca mejor.
 - **Valoración económica:** -
 - **Comentarios:** Puede ser utilizado en compañías que quieran garantizar su independencia y Soberanía Tecnológica.

Modelo 5: Desarrollo de componentes comerciales para productos de software libre.

Ficha 5.1:

- **Modelo:** Desarrollo de componentes comerciales para productos de software libre

- **Autor, Fuente, Fecha:**

1. XOOPS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 [idem, 41]
2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 [idem, 42]

- **Descripción:** Se basa en el desarrollo bajo licencia libre de aplicaciones específicas, en las que se desarrollan determinados componentes comerciales que se distribuyen bajo el modelo de licencias de uso que cubren determinadas necesidades específicas de un cliente. La empresa que lo utiliza es productora, casi en su totalidad, de un producto libre.

- **Aplicado en:**

Ximian: Ha seguido un claro modelo basado en ser la fuente de programas libres. Sus principales productos, como Evolution (es un cliente de correo y agenda de apariencia y funcionalidad similar a Microsoft Outlook.), RedCarpet o Mono se han distribuido bajo licencias de GNU, y Ximian los ha desarrollado desde el comienzo [51]

Zope Corporation: El Modelo de Negocio de la empresa es producir Zope y productos relacionados con él, y ofrecer servicios de adaptación y mantenimiento para todos ellos. Zope Corporation ha sabido, además, crear una dinámica comunidad de desarrolladores de software libre alrededor de sus productos, y colaborar activamente con ellos. [52]

- **Ventajas:** la ventaja competitiva aumenta al ser estas empresas las desarrolladoras del producto en cuestión, controlar su evolución, y tenerlo antes que la competencia. Todo esto la posiciona en un lugar muy bueno de cara a los clientes que deseen servicios sobre ese programa.
- **Desventajas:** para desarrollar estos componentes deben tener sus aplicaciones fundamentalmente sobre plataformas libres.
- **Valoración económica:-**
- **Comentarios:** es un modelo muy interesante en términos de imagen, ya que la empresa puede demostrar su potencial desarrollador con la creación y mantenimiento de la aplicación en cuestión.

Ficha 5.2:

- **Modelo:** Integración de software

- **Autor, Fuente, Fecha:**

1. XOOPS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 [idem, 41]

2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 [idem, 42]
3. Andrea Rossi, et al. Modelos de negocio para SW libre D6.2. Disponible en: http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc 27-02-2007 [idem, 44]
 - **Descripción:** La integración de software consiste en un potente I+D+ i (investigación+ desarrollo+ innovación), capaz de conocer y analizar la mayoría de las herramientas disponibles que se basan en software libre.
 - **Aplicado en:**
 - **openSystems:** es un integrador de soluciones y una empresa española que ha construido un Modelo de Negocio alrededor de la integración de aplicaciones basadas en software libre, ofreciendo soluciones e implantaciones de productos paquetizados en entornos de cliente.
 - **Ventajas:** Con esto es posible construir aplicaciones acorde a las necesidades del cliente mediante la integración de los elementos. El código puede ser rápidamente modificado para adaptarse a nuevas plataformas. Es mucho más flexible
 - **Desventajas:** los desarrolladores en caso de que vayan a desarrollar algo novedoso deben hacer sus previas investigaciones para luego aplicarlas al desarrollo de la aplicación deseada.
 - **Valoración económica:** -
 - **Comentarios:** Utilizando soluciones basadas en software libre se puede partir desde un código existente por lo que no se parte desde cero, donde es posible beneficiarse del proceso de revisión y actualización de versiones de cada proyecto individual basado en software libre sin realizar esfuerzos importantes en reintegrar las modificaciones. Ahorro de grandes inversiones y tiempo.

Modelo 6: Cobrar por el SWL.

Ficha 6.1:

- **Modelo:** Cobrar una mínima cuota por el Software Libre
- **Autor, Fuente, Fecha:**
 - Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [idem, 46]
- **Descripción:** Excepto en casos especiales la licencia GPL no impone ninguna restricción sobre cuanto puedes cobrar por distribuir una copia de Software libre.
- **Aplicado en:** Suse
- **Ventajas:** aplicable a todas las licencias de código abierto y a la que se ciñen muchas compañías

- **Desventajas:** el desarrollador debe de estar seguro de que el código no esté disponible gratuitamente, cuando se disponga a cobrar por el.
- **Valoración económica:** -
- **Comentarios:** este modelo se puede aplicar de diversas formas por ejemplo proporcionando el código fuente a sus clientes junto con los binarios, y cobrándoles por acceder al paquete formado por el código fuente, los binarios y los servicios profesionales. También, vendiendo el software en un CD junto a un manual de usuario, a un precio módico, a muchos le sería más fácil adquirirlo de esta manera que pasarse horas bajando el código.

Ficha 6.2:

- **Modelo:** Liberar la penúltima versión
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-ontent/uploads/2008/04/d62_11.doc. [idem, 44]
- **Descripción:** En este modelo, el software se distribuye bajo una licencia propietaria pero que tenga una fecha de expiración. Por ejemplo se puede crear una licencia que permita la libre distribución, prohíba el uso comercial y garantice que el producto adquiera la licencia GPL después de un año o si el vendedor quiebra
- **Aplicado en:** Aladdin Enterprises
- **Ventajas:** los clientes se aseguran de que el producto es adaptable a sus necesidades ya que disponen del código.
Desventajas: como el código antiguo está liberado puede ser difícil tratar de ver los fallos solucionados, las revisiones y las pequeñas funcionalidades introducidas
- **Valoración económica:** -
Comentarios: Con este modelo la compañía desarrolladora debería obtener más ingresos que distribuyendo el SW con licencias propietarias.

Ficha 6.3:

- **Modelo:** Fuente de un producto con limitaciones
- **Autor, Fuente, Fecha:**
 1. Barahona, Jesús González; Pascual, Joaquín Seoane y Robles, Gregorio. Introducción al software libre. Noviembre 2003. Disponible en http://www.documentacion.prod.hab.uci.cu/Literatura/Cientifica/Maestria_Internacional_SWL [idem, 48]
- **Descripción:** Este modelo tiene 2 limitaciones principales, una es la distribución propietaria durante un tiempo y luego libre donde pasado un tiempo (normalmente, cuando se empieza a comercializar una nueva versión, también

como software propietario), esa versión pasa a distribuirse con una licencia libre. Y la otra es la distribución limitada durante un tiempo aquí el software es libre desde que el productor lo comienza a distribuir durante un tiempo sólo a sus clientes, que le pagan por ello (normalmente en forma de contrato de mantenimiento) y luego lo publica de manera gratuita.

- **Aplicado en:**

Ada Core Technologies: sus productos han sido software libre. Pero la mayoría de ellos los ofrecen primero a sus clientes, como parte de un contrato de mantenimiento. Por ejemplo, su compilador, que sigue estando basado en GCC y se distribuye bajo la GNU GPL, se ofrece a sus clientes como Gnat Pro. [53] **(La segunda limitación)**

- **Ventajas:** para la primera limitación la empresa productora obtiene ingresos de los clientes interesados en disponer lo antes posible de nuevas versiones, y a la vez minimiza la competencia, ya que cualquier empresa que quiera competir usando ese producto sólo podrá hacerlo con la versión libre. Y para la otra el productor obtiene ingresos de sus clientes, que perciben esta disposición preferente del software como un valor añadido.

- **Desventajas:** Debido al retraso con el que el producto está disponible para la comunidad del software libre, es prácticamente imposible que ésta pueda colaborar en su desarrollo, por lo que el productor se beneficia muy poco de contribuciones externas.

- **Valoración económica:** -

Comentarios: el modelo sólo funciona si los clientes a su vez no hacen público el programa cuando lo reciben.

Modelo 7: Venta de SWL por imagen de marcas.

Ficha 7.1:

- **Modelo:** Partnership o franquicias de software libre
- **Autor, Fuente, Fecha:**
 2. MERIT et al. Study on the: Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU. 2006 [54]
- **Descripción:** Una empresa franquicia otorga licencias o marcas, relacionadas con un programa libre determinado.
- **Aplicado en:** ERP Openbravo, la cual franquicia su marca comercial a terceros.
- **Ventajas:** proporciona rápida entrada y salida ya que el franquiciado no tiene que hacer una inversión estableciendo oficinas por el mundo.
- **Desventajas:** deben invertir en publicidad para elevar su imagen.
- **Valoración económica:** -

- **Comentarios:** Este tipo de modelo apoya a las compañías a actuar localmente y pensar globalmente con ayuda de participantes de todo el mundo.

Ficha 7.2:

- **Modelo:** Venta de marca
- **Autor, Fuente, Fecha:**
 1. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2, 27-02-2007. Disponible en: http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc . [idem, 44]
- **Descripción:** es utilizado por empresas que invierten en establecer una marca con buena imagen, y bien reconocida, que más adelante les permita vender con suficiente margen productos libres.
- **Aplicado en:**

Distribución Red Hat Linux: Red Hat ha conseguido posicionar su nombre como el de la distribución de GNU/Linux por excelencia.
- **Ventajas:** puede ofrece una gran cantidad de servicios, tratando de rentabilizar el máximo posible esa imagen de marca
- **Desventajas:** las empresas tienen que invertir en publicidad
- **Valoración económica:** -

Comentarios: Aunque pueden conseguirse productos muy similares por menos dinero, muchos clientes están dispuestos a pagar el extra por comprar una marca.

Modelo 8: Hardware con SWL incluido o embebido.

Ficha 8.1:

- **Modelo:** Hardware con SWL incluido o embebido.
- **Autor, Fuente, Fecha:**
 1. Florian Fainelli, Patrice Kadionik, Thomas Petazzoni y Pierre Ficheux. Sistemas empotrados y Hardware abierto de los RMLL <http://2009.rml.info/> . 2009 [idem, 45]
 2. Laurière Stéphane, et al. Deliverable WdD2.1.3bis State of the art concerning business models for systems comprising open source software. 28/04/2008 Disponible en <http://www.qualipso.org/businessmodels> [idem, 46]
- **Descripción:** se basa en software empotrado en Hardware, mediante la adopción de estándares, de un Kernel estable y de nuevas tecnologías soportadas.

- **Aplicado en:** Neoteris,
Maemo: desarrolló una plataforma creada por Nokia para el Nokia 770, liberaron la propia plataforma de desarrollo bajo una licencia libre, Nokia alienta a que los diseñadores integren todos los componentes teniendo en cuenta los requisitos específicos de los sistemas del hardware. [55]
- **Ventajas:** La combinación de SWL con un hardware específico es también una buena oportunidad para los editores de SWL para crear nuevos productos. Adaptación del núcleo linux a las exigencias de lo empotrado: ninguna unidad de gestión de memoria, poco consumo de memoria, sistemas de ficheros en memoria Flash... y su portabilidad sobre diferentes tipos de microprocesadores.
- **Desventajas:** en ocasiones muchas empresas que se dedican a esto necesitan asociarse a otra que se dedique a esta parte del hardware, ya que ellas no están adentradas en este campo y instalar el SWL en el para luego venderlo. Otra es que los fabricantes de hardware necesitan capacitación en SWL para incluir el SO y componentes a este.
- **Valoración económica:** -
- **Comentarios:** Muchos fabricantes de hardware de computadora han abrazado Linux porque él les permite distribuir el costo del sistema operativo entre múltiples compañías

Modelo 9: Software propietario sobre Software libre.

Ficha 9.1:

- **Modelo:** Software libre como plataforma a software propietario
- **Autor, Fuente, Fecha:**
 1. XOOPS. Copyleft Open Source International Conference. II Conferencia Internacional de Software Libre Conferencia 2005 (article.php.htm), 8/8/2005 10:21:00 [idem, 41]
 2. Fernando Monera Daroqui. Modelos de Negocio basados en Software Libre <http://Opensystem:articuloopen%20Sistemas%20%20Art%C3%ADculo.htm> 08/09/2003 [idem, 42]
 3. Andrea Rossi, et al. Modelos de Negocio para SW libre D6.2. Disponible en: http://www.ines.org.es/vulcano/wp-content/uploads/2008/04/d62_11.doc 27-02-2007 [idem, 44]
- **Descripción:** Este modelo consiste en construir software propietario utilizando herramientas libres o sobre una plataforma libre, y después vender sus servicios, a través de Internet.
- **Aplicado en:**

Google: el modelo va más allá de ser solo software propietario sobre plataforma libre porque si sus desarrollos están patentados, por supuesto su algoritmo de búsqueda lo está, otra gran cantidad de sus desarrollos, como

librerías o como por ejemplo Picasa (herramienta para manejo de fotografías), están liberados e incluso se distribuyen gratis (lo que no hay que confundir con que sean libres). Además, Google mantiene un repositorio donde se publican sus proyectos y librerías liberadas y donde se pueden crear nuevos proyectos libres.

Oracle: han seguido cuidadosamente la adaptación al software libre. Utilizando piezas de código cerradas han intentado (con mayor o menor éxito) la integración con el mundo del software libre tratando de mejorar la compatibilidad y permitiendo la ejecución de su software sobre sistemas operativos libres (como Linux).

- **Ventajas:** Estos se han desarrollado gracias a las plataformas libres que hacen productos sin invertir en costosas plataformas propietarias, ni tener que realizar complicados desarrollos desde su inicio.
- **Desventajas:** Este modelo no logra captar las ventajas competitivas derivadas de Modelos de Negocio puramente basados en software libre por lo que pueden llegar a tener dificultades para obtener rentabilidad a largo plazo.
- **Valoración económica:-**
- **Comentarios:** Como la rentabilidad a largo plazo puede tener dificultades. Y es por esto que hay aplicaciones que necesitan un buen número de usuarios en el mundo del software libre para poder obtener sus ingresos.

Modelo 10: Modelos de Negocio enfocados a las empresas

Ficha 10.1:

- **Modelo:** Empresas que distribuyan el software libre
- **Autor, Fuente, Fecha:**
José Ramón Díaz. Nuevos Modelos de Negocio basados en Software Libre. Diciembre de 2004 Tesina del MBA (UNED) Fundación Guadalquivir [**idem, 17**]
- **Descripción:** una empresa puede crear su propia versión y ofrecer servicios añadidos.
- **Aplicado en:** RedHat comercializa todo tipo de servicios relacionados con la distribución, con GNU/Linux, y con software libre en general. [**56**] El Modelo de Negocio de Red Hat se sustenta en la prestación de un servicio de asistencia técnica profesionalizado a empresas de todo tipo.
- **Ventajas:** pueden crear su propia versión y ofrecer servicios de formación o consultoría, pues cuenta con personal calificado.
- **Desventajas:** algunos productos pueden requerir de una compleja instalación, y numerosas descargas desde Internet.
- **Valoración económica: -**
- **Comentarios:** Red Hat tiene un excelente Modelo de Negocio esta creó su propia versión de Linux facilitando su instalación y administración pero su verdadero negocio está en los servicios de asistencia técnica.

Ficha 10.2:

- **Modelo:** Empresas que den soporte, consultoría o formación al software libre
- **Autor, Fuente, Fecha:**
José Ramón Díaz. Nuevos Modelos de Negocio basados en Software Libre. Diciembre de 2004 Tesina del MBA (UNED) Fundación Guadalquivir [idem, 17]
- **Descripción:** estas empresas tienen equipos bien formados en tecnologías de código abierto que dan servicios y realizan desarrollos a medida de los clientes.
- **Aplicado en:** Red Hat
- **Ventajas:** reducen el precio del sistema operativo (SO) para obtener mayores beneficios.
- **Desventajas:** no todas las empresas dominan el mercado de consultoría y formación sino sólo las distribuidoras.
- **Valoración económica:** -
- **Comentarios:** Esto de dar consultoría y formación es importante pero se necesita de un personal calificado para brindar este servicio a los usuarios.

Ficha 10.3:

- **Modelo:** Empresas que se apoyen en programas de SCA
- **Autor, Fuente, Fecha:**
José Ramón Díaz. Nuevos Modelos de Negocio basados en Software Libre. Diciembre de 2004 Tesina del MBA (UNED) Fundación Guadalquivir
- **Descripción:** determinadas empresas renuevan sus productos para que funcionen sobre plataformas de código abierto con el fin de independizarse de empresas proveedoras.
- **Aplicado en:** SAP, Oracle, Hewlett-Packard o Dell Computer
- **Ventajas:** pueden aumentar sus márgenes, y competir en el mercado, donde sus posiciones son más ventajosas.
- **Desventajas:** estas empresas para poderse basar en SCA deben migrar primero para poder independizarse
- **Valoración económica:** -
- **Comentarios:** este modelo es una alternativa viable para aquellas empresas que quieran comercializar sus productos en el mercado.

Ficha 10.4:

- **Modelo:** Empresas que utilizan programas SCA como usuarios finales.
- **Autor, Fuente, Fecha:**

José Ramón Díaz. Nuevos Modelos de Negocio basados en Software Libre. Diciembre de 2004 Tesina del MBA (UNED) Fundación Guadalquivir

- **Descripción:** Estas empresas realizan modificaciones necesarias para ellas en el software y luego compartirlas con la comunidad para aportar un valor añadido al software libre., pero
- **Aplicado en:**
- **Ventajas:** lo utilizan porque les aporta un mayor valor añadido, o bien por que aumenta sus márgenes operativos. Pueden en cualquier momento aportar modificaciones a la comunidad.
- **Desventajas:** no generan ingresos por el software libre
- **Valoración económica:-**
- **Comentarios:** es un modelo que no aporta ningún beneficio en lo que respecta a su economía. No aplica

Ficha 10.5:

- **Modelo:** Empresas de venta de accesorios para el software libre
- **Autor, Fuente, Fecha:** José Ramón Díaz. Nuevos Modelos de Negocio basados en Software Libre. Diciembre de 2004 Tesina del MBA (UNED) Fundación Guadalquivir
- **Descripción:** generan un negocio satélite de venta de libros, CD o documentación.
- **Aplicado en:** Red Hat
- **Ventajas:** obtienen sus ingresos por esta vía.
- **Desventajas:** en algún momento estos accesorios pueden perder su valor cuando pasa un tiempo determinado.
- **Valoración económica:-**
- **Comentarios:** este modelo puede generar ingresos por varios métodos, donde siempre va existir un cliente que necesite estos servicios.

Ficha 10.6:

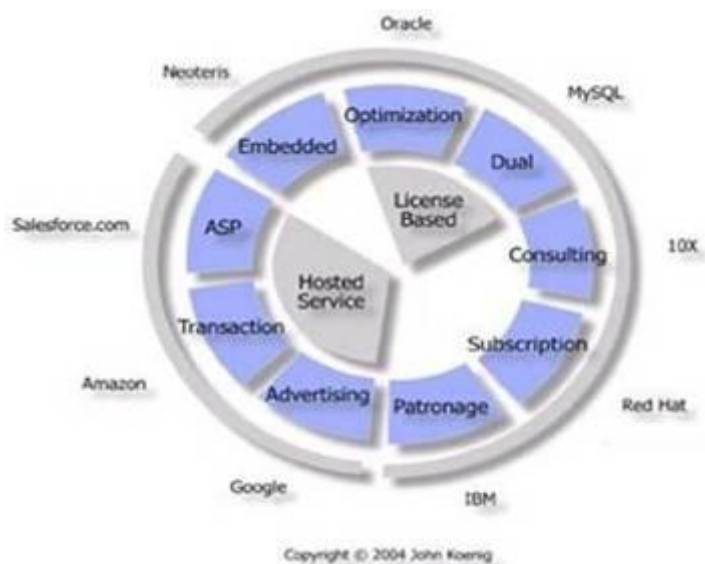
- **Modelo:** Modelo de Negocio basado en la filosofía del Software Libre para una Empresa Cubana de Soporte.
- **Autor, Fuente, Fecha:** Marcos Luís Ortiz Balmaceda. Proyecto Unicornios. mlortiz@estudiantes.uci.cu. Revista de Software libre. /etc/init.d/uxi start/No. 08 Vol. 02/, Tomado el 5 de marzo del 2009 [57]
- **Descripción:** creación de una empresa nacional que brinde soporte al software libre a nivel no sólo nacional sino también a las empresas, ministerios o países del área que requieran de dichos servicios. Esta empresa atendería las siguientes aristas: el diseño y desarrollo Web basados en un excelente manejador de contenido (CMS Drupal), apoyo al desarrollo de la distribución nacional creada en la Universidad de las Ciencias Informáticas (NOVA Linux),

desarrollo de una distribución nacional específicamente para servidores basada en OpenBSD, creación de un grupo de desarrollo bien preparado y comprometido en la constante mejora del RDBMS PostgreSQL.

- **Aplicado en:** para ser aplicado en una empresa nacional
- **Ventajas:** múltiples ventajas sobre otros sistemas operativos en cuestiones de seguridad y criptografía aplicada, temas muy sensibles hoy en día en el mundo informático; puede llegar a convertir al país en una potencia de desarrollo de este mismo software, para que pueda competir con los mejores sistemas de bases de datos comerciales del mundo, además de ser uno de los grupos de desarrollo más grandes en América Latina.
- **Desventajas:** modelo limitado, solo puede usarse como CMS, Drupal. Es muy costoso y lleva mucho tiempo porque no es lo mismo crear una empresa que un software.
- **Valoración económica:-**
- **Comentarios:** Este modelo debe ir más allá que a una empresa, es decir no solo limitarlo a esta sino a que pueda ser usado por otros y sin restricciones para que puedan comercializar el OSS (Open Source Software) de una forma ética y profesional.

Anexo 2: Modelos según IT Managers Journal.

Fuente: Kobayashi Gutiérrez, Alfred. El Modelo de Negocio del Software Libre, como eje en la Industria de Software Peruano para competir en el mercado Mundial 10/21/06. Tomado el 8 de febrero del 2009. [idem, 16]



- 1) **La Estrategia de Optimización:** Se refiere a intentar mantener modularidad.



ORACLE™

- 2) **La Estrategia de Licencia Dual :** Desarrollo de Software bajo dos tipos de Licencia



- 3) **La Estrategia de Consultoría:** Se refiere a los servicios de las empresas por la personalización (customización) de un software o del desarrollo de un software a medida.



4) La Estrategia de Suscripción



5) La Estrategia del Patrocinio



6) La Estrategia de Almacenamiento



7) La Estrategia Embebida

Software embebido en Hardware

- Adopción de estándares
- Kernel Estable
- Nuevas tecnologías soportadas

Anexo 3: Algunos proyectos de SWL que existen actualmente en la UCI.

Fuente: Ing. Abel Meneses Abad. Migración ¿Qué proyectos usan y desarrollan Software Libre en la UCI? Marzo 2007. Revista de SWL de la UCI. Consultado 2 marzo 2009 y actualizado en abril del 2009. [58]

Algunos proyectos de SWL existentes en la UCI					
No	Nombre	Descripción	Contactos	Tecnologías	Áreas
1	Nova	Desarrollo de una distribución libre para la República de Cuba.	Angel Goñi Oramas agoni@uci.cu	Gentoo, C++, GTK	Polo de SWL
2	Filpacon	Desarrollo de un filtro de Internet para Cuba, y empresas.	Luis Enrique Sánchez lesanchez@uci.cu	Debian, Perl, PHP, MySQL	Polo de Soluciones Informáticas de Internet
3	RINDE	Red de Integración y Desarrollo, una red para el trabajo, la promoción del SWL en Venezuela	Juan Manuel Pernia Rodríguez jpernia@uci.cu	gforge, wiki	Polo de SWL
4	Fábrica de Portales	Desarrollo de portales Web para la intranet nacional	Maikel Manuel Fernández Fernández	Drupal/PHP & Zope/Plone/Python	Facultad 10

			maikelm@uci.cu		
5	Proyecto UNICORNIOS (Servicios Especializados para la Migración a Software Libre)	Soporte, y desarrollo de aplicaciones, para la migración a SWL.	Abel Meneses Abad abelma@uci.cu	C++, PHP, Python, Perl, Bash,	Polo de SWL
6	SCADA Nacional PDVSA	Desarrollo de un Sistema Supervisor de Procesos Automatizados únicamente con tecnología libre.	Juan Antonio Fung Goizueta fung@uci.cu	Java	Polo de Hardware y Automatización
7	Intranet PDVSA	Desarrollo de una intranet corporativa	Cesar González Hernández	PHP	Facultad 10
8	PostgreSQL	Este desarrolla un paquete de herramientas para el soporte y la extensión del gestor PostgreSQL que incluye el montaje de clusters de bases de datos sobre este gestor, soporte al diseño y la modelación de bases	Pedro Yobanis Piñero Pérez ppp@uci.cu	PostgreSQL,	CENTALAD, UCI

		de datos, entre otras cosas.			
9	Entrenadores Aduana	El proyecto surge como una iniciativa de la directiva de la AGR (Aduana General de la República) para mejorar el entrenamiento y la preparación de los inspectores aduaneros que cada año se forjan en la ENFA (Escuela Nacional de Formación Aduanera).	Jaime González Campistruz jgonzalezc@uci.cu	ORACLE	Polo de Realidad Virtual
10	El Archivo General de la Nación de la República Bolivariana de Venezuela (AGN)	Tiene la misión de preservar y difundir la documentación que representa la memoria histórica del país, haciendo uso de estándares abiertos y de las Tecnologías de Información y Comunicación (TICs) para lograr la digitalización del Fondo Documental.	Adrián Cid Almaguer adriancid@uci.cu	Mysql PHP Apache CodeIgniter	Polo de Gestión de la Información y el Conocimiento en la Facultad 10.

Anexo 4: La GPL y su compatibilidad entre licencias

Fuentes: [59] Compatibilidad entre licencias. Disponible en <http://producingoss.com/es/license-aspects.html>. Consultado en abril del 2009 y [idem, 5] Diversas licencias y comentarios sobre ellas. <http://www.gnu.org/licenses/license-list.html> Consultado 29/1/2009

La compatibilidad con la GPL es un problema importante cuando se elige una licencia. Si se quiere que un código se pueda emplear libremente con código GPL se debe elegir una licencia compatible con la GPL. Algunas de las licencias compatibles con la GPL son también compatibles con software propietario. Por supuesto, los **resultados** de estas mezclas no serán compatibles con la otra, ya que una estará bajo la GPL y otra estará bajo una licencia de código cerrado. Esa preocupación se aplica únicamente a las obras derivadas, y no al código que se distribuya en primer lugar.

La Free Software Foundation mantiene una lista que muestra qué licencias son compatibles con la GPL y cuáles no:

Licencias de software libre compatibles con GPL

- **La Licencia Pública General de GNU, o GNU GPL:** Es una licencia de software libre, y de tipo copyleft. Recomendada para la mayoría de los paquetes de software.
- **La Licencia Pública General Reducida de GNU, o GNU LGPL:** Es una licencia de software libre, pero no tiene un copyleft fuerte, porque permite que el software se enlace con módulos no libres. Se recomienda para circunstancias especiales.
- **La licencia de Guile:** Consiste en la GNU GPL más una declaración especial que permite enlazar con software no libre. No tiene un copyleft fuerte, pero es compatible con la GNU GPL. Se recomienda sólo para casos especiales, que suelen ser los mismos en los que se puede considerar el uso de la LGPL.
- **La licencia X11:** Es una licencia de software libre simple y permisiva sin copyleft pero compatible con la GNU GPL. XFree86 usa la misma licencia. A veces se le llama la licencia del "MIT", pero ese término es engañoso puesto que el MIT ha utilizado muchas licencias para su software.
- **La licencia Expat:** Es una licencia de software libre simple y permisiva sin copyleft pero compatible con la GNU GPL. A veces se la cita de forma ambigua como la Licencia MIT.
- **La Licencia de Copyright ML Estándar de New Jersey:** Es una licencia de software libre simple y permisiva sin copyleft pero compatible con la GNU GPL.

- **La Licencia General de Cryptix:** Es una licencia de software libre simple y permisiva sin copyleft pero compatible con la GNU GPL. Es muy similar a la licencia X11.
- **La licencia BSD modificada:** Es la licencia BSD original, modificada por la eliminación de la cláusula de publicidad. Es una licencia de software libre simple y permisiva sin copyleft pero compatible con la GNU GPL. Se recomienda como alternativa la licencia X11. Esta licencia y la BSD modificada son más o menos equivalentes.
- **La licencia de ZLib**
- **La licencia de la "Biblioteca de Funciones Estándar" de iMatix**
- **El aviso y la licencia de software del W3C**
- **La licencia de la base de datos de Berkeley** (también llamada licencia de productos software de Sleepycat)
- **La licencia de OpenLDAP, versión 2.7:** Es una licencia de software libre permisiva y sin copyleft pero compatible con la GNU GPL.
- **La Licencia de Python, versión 1.6a2 y anteriores**
- **La Licencia de Python, versiones 2.0.1, 2.1.1 y posteriores:** las versiones intermedias de Python (de la 1.6b1 a la 2.0, más la 2.1) están bajo otras licencias (vea más abajo).
- **La licencia de Perl:** Esta licencia plantea una disyuntiva entre la Licencia Artística y la GNU GPL. Dicho de otro modo, se puede elegir entre cualquiera de las dos licencias. Es una licencia de software libre pero puede no ser realmente de tipo copyleft. Se recomienda, utilizar esta licencia para cualquier paquete de Perl 4 ó Perl 5 que escriba, para promover coherencia y uniformidad en la programación en Perl. Fuera de Perl, no utilice esta licencia; es mejor utilizar sólo la GNU GPL.
- **La Licencia Artística con aclaraciones:** Contiene el mínimo conjunto de cambios necesarios para corregir la vaguedad de la Licencia Artística Original.
- **La Licencia Artística 2.0:** ya no está en uso; pero está siendo considerada en Perl 6 como parte de su esquema de licencia dual.
- **La Licencia Pública de Zope versión 2.0:** Es una licencia de software libre simple y permisiva sin copyleft pero compatible con la GNU GPL.
- **La Licencia de Código Abierto de Intel** (según la "Open Source Initiative"): Es una licencia de software libre compatible con la GNU GPL.
- **La licencia de Javascript de Netscape:** Es una disyuntiva entre la Licencia Pública de Netscape y la GNU GPL, es una licencia de software libre compatible con la GNU GPL, pero no tiene un copyleft fuerte.

- **La licencia eCos versión 2.0:** Consiste en la GPL más una excepción que permite enlazar con software que no está bajo la GPL. Esta licencia tiene los mismos inconvenientes que la LGPL.
- **La licencia del Forum Eiffel, versión 2:** las versiones anteriores de la licencia Eiffel no son compatibles con la GNU GPL.
- **La licencia de Vim, versión 6.1 o posterior:** Es una licencia de software libre, parcialmente copyleft aunque no totalmente. Es compatible con la GPL por la existencia explícita de una cláusula de conversión.

Licencias de software libre incompatibles con la GPL

- **La Licencia Pública General de Affero:** es una licencia de software libre, de tipo copyleft, pero no es compatible con la GNU GPL. Consiste en la GNU GPL versión 2, con una sección adicional que Affero añadió con la aprobación de la FSF.
- **La Licencia Pública Arphic:** Es una licencia de software libre de tipo copyleft, pero no es compatible con la GNU GPL. Su uso habitual es para fuentes tipográficas, y para ese uso la incompatibilidad no constituye un problema.
- **La licencia BSD original:** Es una licencia de software libre simple y permisiva sin copyleft con un grave defecto: la irritante "cláusula publicitaria de BSD". El defecto no es fatal ya que el software sigue siendo libre, pero provoca problemas prácticos, en particular, incompatibilidad con la GNU GPL.*
- **Common Development and Distribution License (CDDL):** Es una licencia de software libre. Tiene un esquema de copyleft similar al de la Licencia Pública de Mozilla, lo que la hace incompatible con la GPL de GNU. Esto significa que un módulo cubierto por la GPL y un módulo cubierto por la CDDL, no pueden estar enlazados legalmente.
- **La licencia de OpenSSL:** es una combinación de dos licencias: la propia licencia de OpenSSL y la de SSLeay. Han de cumplirse ambas. La combinación tiene como resultado una licencia de software libre de tipo copyleft incompatible con la GNU GPL. Tiene una cláusula de publicidad al igual que la licencia BSD original y la licencia de Apache.
- **La Licencia Libre Académica (AFL), versión 1.1:** Es una licencia de software libre sin copyleft e incompatible con la GNU GPL en varios aspectos. La presentan como una "actualización compatible" para

* no use la licencia BSD original para el software que escriba. Si quiere usar una licencia simple y permisiva sin copyleft, es mucho mejor usar la licencia BSD modificada o la licencia X11. Sin embargo no hay razón para no usar programas que hayan sido distribuidos bajo esta licencia.

"licencias tales como la BSD y la MIT", pero no es así; la licencia BSD revisada y la licencia MIT son compatibles con la GPL, pero la AFL no lo es.

- **La Licencia de Software Abierto (OSL), versión 1.0:** es una licencia de software libre. Sus autores dicen que se trata de una licencia de tipo copyleft, pero encontramos dificultades para determinar si las garantías de copyleft indicadas son válidas. Es incompatible con la GNU GPL en varios aspectos.
- **La Licencia de Apache, versión 1.0:** Es una licencia de software libre simple y permisiva sin copyleft con problemas prácticos como los de la licencia BSD original, incluyendo incompatibilidad con la GNU GPL.
- **La Licencia de Apache, versión 1.1:** Es una licencia de software libre permisiva y sin copyleft con algunos requisitos que la hacen incompatible con la GNU GPL.
- **La Licencia de Apache, versión 2.0:** Es una licencia de software libre pero incompatible con la GPL porque tiene un requisito concreto que no tiene la GPL: contempla algunos casos, que la GPL no requiere, en los que puede rescindirse la licencia por problemas de patentes.
- **La Licencia Pública de Zope versión 1:** Es una licencia de software libre simple, bastante permisiva y sin copyleft con problemas prácticos como los de la licencia BSD original, incluyendo incompatibilidad con la GNU GPL. **
- **La licencia de xinetd:** Es una licencia de software libre de tipo copyleft incompatible con la GNU GPL. Es incompatible porque pone restricciones adicionales en la redistribución de las versiones modificadas que contradicen los requisitos de la GPL.
- **La Licencia de Python 1.6b1 y versiones posteriores hasta la 2.0, más la 2.1:** Es una licencia de software libre pero es incompatible con la GNU GPL. La principal incompatibilidad es que estas licencias están sujetas a las leyes del estado de Virginia, en los EE.UU., y la GPL no permite tal cosa.
- **La antigua licencia de OpenLDAP, versión 2.3:** Es una licencia de software libre permisiva con algunos requisitos (en las secciones 4 y 5) que la hacen incompatible con la GNU GPL. la última versión de OpenLDAP tiene una licencia diferente que es compatible con la GNU GPL.
- **La Licencia Pública de IBM, versión 1.0:** Es una licencia de software libre pero es incompatible con la GPL. Ya que, requiere que se den ciertas licencias sobre las patentes, algo que la GPL no hace.
- **Licencia Pública Común versión 1.0:** Es una licencia de software libre pero es incompatible con la GPL porque requiere que se den ciertas licencias sobre las patentes, algo que la GPL no hace.
- **La licencia de Phorum, versión 2.0:** Es una licencia de software libre pero es incompatible con la GPL. La sección 5 es la que provoca dicha incompatibilidad.

** La última versión de Zope está disponible bajo una licencia compatible con GPL.

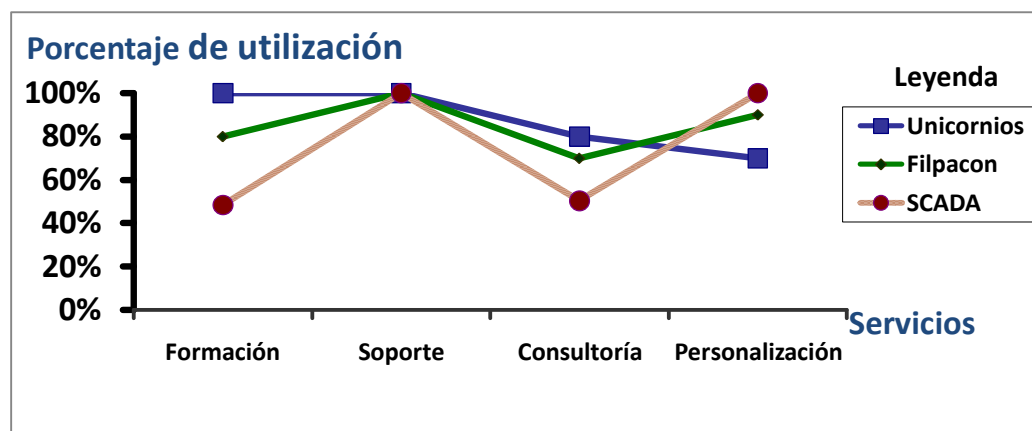
- **La Licencia Pública del Proyecto LaTeX (LPPL):** Esta licencia no contiene todos los términos de la distribución de LaTeX. Esta licencia contiene restricciones complejas y molestas sobre cómo publicar una versión modificada, incluyendo un requisito que la sitúa en el límite de lo aceptable: el hecho de que cualquier archivo modificado debe ser renombrado. Por favor, no use esta licencia en ningún otro proyecto.

- **La Licencia Pública de Mozilla (MPL):** Es una licencia de software libre, pero no tiene un copyleft fuerte. A diferencia de la licencia X11, tiene algunas restricciones complejas que la hacen incompatible con la GNU GPL. De hecho, no se puede, legalmente, enlazar un módulo cubierto por la GPL con un módulo cubierto por la MPL.
- **La Licencia de Código Fuente Abierto de Netizen (NOSL), versión 1.0:** Es una licencia de software libre, que es esencialmente igual a la Licencia Pública de Mozilla, versión 1.1. Al igual que ésta, la NOSL tiene algunas restricciones complejas que la hacen incompatible con la GNU GPL. De hecho, no se puede, legalmente, enlazar un módulo cubierto por la GPL con uno cubierto por la NOSL.
- **La Licencia Pública de Interbase (IPL), versión 1.0:** Es una licencia de software libre que es esencialmente igual a la Licencia Pública de Mozilla, versión 1.1. Al igual que ésta, la IPL tiene algunas restricciones complejas que la hacen incompatible con la GNU GPL. De hecho, no se puede, legalmente, enlazar un módulo cubierto por la GPL con uno cubierto por la IPL.
- **La Licencia Pública de Sun:** Es esencialmente igual a la Licencia Pública de Mozilla: una licencia de software libre incompatible con la GNU GPL. Por favor, no la confunda con la Licencia "Sun Community Source" ya que ésta no es de software libre.
- **La Licencia de Código Fuente Abierto de Nokia:** Es similar a la Licencia Pública de Mozilla: una licencia de software libre incompatible con la GNU GPL.
- **La Licencia Pública de Netscape (NPL):** Es una licencia de software libre, sin un copyleft fuerte, e incompatible con la GNU GPL. Consiste en la Licencia Pública de Mozilla con una cláusula añadida que permite a Netscape usar el código que usted añada incluso en sus versiones privativas del programa. Por supuesto usted no tiene derecho a usar el código de Netscape en contrapartida.
- **La Licencia de Código Fuente Abierto de Jabber, versión 1.0:** Es una licencia de software libre incompatible con la GNU GPL. Permite redistribuir el código bajo cierta clase de licencias: aquellas que incluyen todos los requisitos de la licencia de Jabber. Como la GPL no pertenece a esa clase, la licencia de Jabber no permite redistribuir el código bajo la GPL. De ahí viene su incompatibilidad.

*** Nota: Estos comentarios están hechos sobre la versión 1.2 (3 de septiembre de 1999) de la LPPL.

- **La Licencia para el Código Fuente según los Estándares Industriales de Sun, versión 1.0:** Es una licencia de software libre, sin un copyleft fuerte, e incompatible con la GNU GPL por pequeños detalles más que por su política principal.
- **La Licencia Pública Q (QPL), versión 1.0:** Es una licencia de software libre sin copyleft e incompatible con la GNU GPL. También es causa de importantes inconvenientes prácticos, pues las fuentes modificadas sólo pueden distribuirse como parches.
- **La licencia FreeType:** Es una licencia de software libre sin copyleft, incompatible con la GNU GPL por razones técnicas.
- **La Licencia de PHP, versión 3.0:** Esta es la licencia usada por la mayor parte de PHP4. Es una licencia de software libre sin copyleft, incompatible con la GNU GPL.
- **La Licencia Zend, versión 2.0:** Esta licencia se usa en parte de PHP4. Es una licencia de software libre sin copyleft con problemas prácticos como los de la Licencia BSD Original.
- **La Licencia de Plan 9 (junio de 2003):** Es una licencia de software libre incompatible con la GNU GPL. Se recomienda, que no use esta licencia para el software que escriba, pero es aceptable usar y mejorar Plan 9 bajo esta licencia.
- **La Licencia de Fuente Pública de Apple (APSL), versión 2:** Es una licencia de software libre incompatible con la GNU GPL. Se recomienda, que no use esta licencia para el software que escriba, pero es aceptable usar y mejorar el software distribuido bajo esta licencia. Están disponibles más explicaciones.

Anexo 5: Gráfica sobre porcentaje de utilización de los servicios ofrecidos por los Proyectos tomados como Casos de Estudio en el Capítulo 3.



Nota: Esta gráfica muestra el porcentaje de utilización de los servicios que estos proyectos ofrecen. En los de Formación, Soporte y Consultoría, UNICORNIOS presenta el de mayor porcentaje porque es un proyecto que está orientado a la comercialización de estos. En el caso del Soporte todos los proyectos presentan el máximo ya que a la hora de comercializar un producto se le ofrece este tipo de servicio para su buen funcionamiento, pero en cuanto a la personalización SCADA es el que más porcentaje presenta, porque es un proyecto que se dedica a desarrollar aplicaciones a la medida y a partir de ellas es que se orienta su comercialización en el mercado.

Anexo 6: Encuesta para conocer el grado de participación que tienen los desarrolladores en las diferentes comunidades de SWL.

1. ¿Has participado en alguna comunidad asociada al SWL?

____ Sí ____ No

¿Cuál/es?

¿Cómo te integraste?

¿En qué medida contribuyes?

¿Has sido reconocido? Sí ____ No ____

¿Cómo?

La encuesta anterior fue realizada a 27 desarrolladores de diferentes proyectos productivos dedicados al SWL en la Universidad de las Ciencias Informáticas, siendo estos: Proyecto SCADA, Diferentes proyectos del Grupo UNICORNOS, Fábrica de Portales, Proyecto Aires, Proyecto de Calidad, Proyecto GeWeb; para conocer por esta vía el grado de

participación de sus desarrolladores en las diferentes comunidades y en qué medida se integran, contribuyen y son reconocidos.