

# Alternativas y Razones para el uso de Software de libre adquisición, en la práctica docente de las ingenierías

Aquino Robles, José Antonio<sup>1 2 3</sup>, Corona R., Leonel G.<sup>2</sup> Trujillo Caballero Juan Carlos<sup>4</sup>

## Nota sobre los autores

<sup>1</sup> Departament d' Enginyeria Elèctrica, Universitat Politècnica de Catalunya, jaquinor@ipn.mx

<sup>2</sup> Academia de Mecatrónica, Unidad Profesional Interdisciplinaria de Ingeniería y Tecnologías Avanzadas

<sup>3</sup> Becario de la Fundación Carolina, del Instituto de Investigaciones Eléctricas IIE y de COTEBAL IPN

<sup>4</sup> Departamento de Ingeniería Eléctrica, Instituto Tecnológico de Orizaba, México

Aquino, J., Corona, L. y Trujillo, J. (2014) Alternativas y Razones para el uso de Software de libre adquisición, en la práctica docente de las ingenierías. *Espacio I+D Innovación más Desarrollo*, 3 (4), 26-45. doi: 10.31644/IMASD.4.2014.a03

## Resumen

Actualmente se vive una modernización educativa, la cual está plasmada en los Modelos Educativos de Vanguardia. En las Instituciones de Educación Superior (IES) se propone en su discurso la incorporación masiva de las Tecnologías de la Información y la Comunicación (TIC). Mismas que se pretende utilizar en todas las carreras y en especial dentro de la enseñanza en las aulas. Por tales hechos; esta propuesta expone y describe alternativas y razones para utilizar software de libre adquisición y distribución. Disponibles para su uso en la enseñanza y fomentar el aprendizaje de Ingeniería en México.

Basados en la ordenanza de no sólo incentivar el uso masivo de las TIC, sino hacerlo de manera ética y honesta en la práctica docente. Se opta por promover el uso de dichas herramientas. Considerando que el factor del costo del software propietario, en caso de no disponer de los recursos económicos, llega a ser difícil de adquirir legalmente y por ello motivar prácticas que contradicen los códigos de ética del desempeño profesional.

*Palabras Clave: Software Libre, Software de código fuente abierto, Enseñanza, Aprendizaje, Ingeniería.*

## Abstract

Currently, we are experiencing a modernization of education embodied in cutting edge educational models. In the institutions of higher education is proposed in his speech the massive incorporation of the information technologies and communication. Which seek to utilize in their careers and especially in the classroom teaching. For these facts, in this proposal we seeks to identify and describe free software tools acquisition and distribution, available for use in teaching and supporting learning in Engineering in Mexico. Based on the Ordinance not only encourage the widespread use of TIC, but do so ethically and honestly in the teaching practice. It was decided to promote the use of these tools. Whereas the cost factor of proprietary software, if you do not have economic resources becomes difficult to acquire legally and therefore encourage practices that contradict the codes of ethics of professional performance.

*Keywords: Free Software, Open Source Software, Teaching, Learning, Engineering.*

## Introducción

Existe dentro de las iniciativas del Modelo Educativo institucional del Instituto Politécnico Nacional la de incorporar las Tecnologías Informáticas y de Comunicación (TIC) al trabajo académico a todos los niveles educativos de esta institución, promoviendo el uso de software en el aula, como apoyo didáctico y con ello originar aprendizajes significativos en los estudiantes. Por lo que respecta a la propuesta del Modelo Educativo (ME) del Instituto Politécnico Nacional, se señala que el IPN deberá distinguirse porque sus servicios sean de la más alta calidad y que respondan al reto de crear más oportunidades de educación para los jóvenes, enfocándose en seis objetivos fundamentales:

1. Centrado en el aprendizaje
2. Flexibilidad y atención en el alumno
3. Integración de lo científico, tecnológico y humanístico
4. Internacionalización y multiculturalidad
5. Autonomía del aprendizaje
6. Concepción renovada del vínculo entre lo teórico y lo práctico

## Desarrollo

Si nos centramos en las TIC dentro del ámbito de la docencia, la utilización de software está cada vez más extendida. En las Universidades, y concretamente en las asignaturas de ciencias de la ingeniería y más aún en las de tecnología aplicada, que se imparten en los planes de estudio de Ingeniería, se utiliza con frecuencia programas que realizan diversos cálculos y representan funciones. Sin embargo, los precios de las licencias de uso de estas herramientas, pueden limitar su utilización a las instituciones y con mayor motivo en el alumnado. (Un Nuevo Modelo Educativo para el IPN 2003)

Considerando que en este tiempo el Software de libre adquisición ha cobrado ya gran importancia y cada vez son más los usuarios que lo manejan. Hace unos años quizá este tipo de software era reducto de

fanáticos informáticos. Hoy cualquiera ha oído hablar, e incluso usa, programas como los navegadores web *Firefox* u *Opera* etc., de acuerdo a la definición presentada por Rodríguez R. M. 2011; un programa es software libre si el usuario tiene las siguientes libertades o derechos:

- a. La libertad de ejecutar el programa, con cualquier propósito.
- b. La libertad de estudiar la forma cómo funciona el programa, y adaptarlo a sus necesidades.
- c. La libertad de distribuir copias.
- d. La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad usuaria se beneficie.

Existen varias licencias que garantizan que los programas distribuidos como software libre se mantengan como tal, siendo la más común la GNU GPL<sup>1</sup>. A los programas, cuyas licencias privan del derecho a estudiarlos, modificarlos o redistribuirlos, se les conoce como Software Propietario, siendo este último por definición aquel en el que un usuario tiene limitadas sus posibilidades de usarlo, modificarlo o redistribuirlo, y las más de las veces su licencia tiene un coste (Stallman. R. M. 2004).

Como ya se mencionó el ME del IPN reclama nuevas herramientas para lograr los seis objetivos anteriormente enlistados y el uso de distintos paquetes de Software de libre adquisición, posibilita la realización de dichos objetivos de una forma sencilla, ética y económica para los alumnos e instituciones. En este contexto el uso de aplicaciones de Software libre, de Software de libre adquisición y de código fuente abierto<sup>2</sup> puede ayudar decisivamente a la renovación

<sup>1</sup> GPL significa General Public License (Licencia Pública General).

GNU es un proyecto que se inició en 1984. GNU es un acrónimo recursivo que literalmente significa "GNU No es Unix".

<sup>2</sup> **Software libre, Software de libre adquisición y el de código fuente abierto** - No son la misma cosa, aunque han nacido los tres casi a la par. El movimiento Código fuente abierto (Open Source en inglés) se crea por una sencilla razón, en inglés "free" significa libre y gratis a la vez, y mucha gente cuando veía "Free Software" (que es la traducción de Software Libre) lo confundía con "Software Gratis", siendo cosas totalmente diferentes. El Software de libre distribución resulta ser en el mayor de los casos software privativo que se suele ofrecer gratuitamente en versiones con recursos limitados para cautivar parte del mercado, en su mayoría se conoce como la versión estudiantil del software.

pedagógica que está incluida en la adaptación a los seis objetivos fundamentales, enumerados anteriormente. Los alumnos tendrán a su disposición (de forma gratuita y desde casa) las mismas herramientas que el profesor. Lo que les hará mucho más fácil involucrarse de forma activa y creativa (lo cual no es poco) en el proceso enseñanza-aprendizaje de la(s) asignatura(s). Y si al mismo tiempo consideramos dentro de un enfoque holístico, y por ende integral en todos los sentidos, que como docentes no solo transmitimos conocimientos científicos, habilidades y saberes tecnológicos, sino que al hacerlo debemos tener el cuidado de llevarlo a cabo de forma ética y con la honestidad propia de un profesional de la enseñanza. En este aspecto no sólo estaríamos enseñando a trabajar de forma ética en lo teórico sino también en lo práctico.

## Desarrollo

Respecto a lo anterior, al considerar el uso del software de libre distribución, no sólo debe tenerse en cuenta los aspectos: ético y de gratuidad de la mayoría de ellos, a continuación se enlistan una serie de atributos que se considera necesario mencionar:

- **Atributos Económicos.** Con las herramientas informáticas o software propietario es necesario adquirir licencias para su uso. Por ejemplo, muchos profesores universitarios emplean programas matemáticos de cálculo simbólico (CAS) las cuales no hace falta mencionarlas para el fin de esta investigación. Estos programas tienen un costo que a decir verdad es poco probable que pueda pagarlo un estudiante universitario o preuniversitario, si pretenden trabajar con ellos en casa. Con todo recordemos el dicho que resume la ética de GNU: "El software libre es una cuestión de libertad, no de precio" (Bayón L., Grau J.M., Otero J.A., Ruiz M.M., Suárez P.M 2011).
- **Atributos Legales.** Una prerrogativa de usar software de libre distribución y adquisición en la docencia, es que podemos distribuir copias del programa legalmente a los alumnos. El uso de programas libres evita tener que acudir a trabajar con programas con copias ilegales. La licencia del programa libre nos autoriza por antonomasia su libre distribución: "Si empleas un software sin la autonomía de distribuir copias, tarde o temprano te verás ante un aprieto

moral cuando un camarada te diga: Ese programa que tienes me sería útil. ¿Podrías darme una copia?” (Ibíd).

- **Atributos Científicos.** Se percibe una estrecha correlación entre el uso del software libre y el Método Científico. En efecto, en la ciencia es fundamental la crítica de los resultados por parte de otros científicos, para de esta forma dar con nuevos resultados. Esencialmente lo mismo ocurre en el universo del software libre. Uno puede tomar el programa de otro usuario, calarlo, renovarlo, mejorarlo, optimizarlo, y además utilizarlo como fundamento para ampliar un nuevo programa u otra aplicación.
- **Atributos Formativos.** Cuando se adiestra a los estudiantes con fundamentos puntuales en ciertos programas concretos, se cae en la desventaja de que estos, con el paso del tiempo, queden arcaicos, más ahora que se renuevan constantemente. Además la formación de los alumnos con este tipo de software es independiente de una tecnología concreta. Se busca, por tanto de una *formación basada en fundamentos y no en aplicaciones*.
- **Atributos Filosóficos.** Si partimos con el fundamento de que la formación en ingeniería no es solamente transferir una serie de conocimientos y habilidades técnicas o prácticas, sino que se busca también trasladar valores positivos en lo social, resulta claro que el uso de este tipo de software contribuye a este propósito. Utilizando programas de software libre se promueven en el salón de clases, valores como la igualdad de oportunidades, la libertad, el conocimiento compartido, la creatividad, la solidaridad, el respeto a las leyes “Un programa que sea software libre puede quizás no ser técnicamente superior, pero siempre será éticamente superior” (Ibídem).

Es fundamental entender que con el concepto de software libre no estamos hablando simplemente de software gratuito: el software libre se puede vender si se desea (y en muchas ocasiones puede haber quien esté interesado en pagarlo). En este sentido, el software libre también puede convertirse en “software comercial” y, de hecho, parte del modelo de negocio de algunas empresas (por ejemplo, distribuidoras de GNU/Linux como RedHat o SuSE) se centra en la venta de software libre. Aunque quien lo adquiera debe ser consciente de que (debido a la tercera libertad) podrá redistribuirlo

cuando desee y como lo desee, por ejemplo sin pedir dinero a cambio, ni permiso a alguien.

También es conveniente distinguir el software libre de otros conceptos, como freeware (software gratuito, pero sin libertad de estudio o modificación) o shareware (programas que se pueden evaluar de forma gratuita pero durante un tiempo, uso o características limitadas). A nivel práctico, asociaremos el concepto de software libre con el de "software de fuente abierta" o de "código abierto" (open source), pues la única distinción es el enfoque que normalmente desean transmitir quienes utilizan esta denominación, mucho más pragmático.

## Desde el punto de vista de la Ética

Conforme al diccionario de la Real Academia de la lengua Española la Ética es:

- El segmento de la filosofía que concierne a la moral y de los deberes del hombre.
- El conjunto de pautas morales que tutelan la conducta humana.

Sí consideramos que la Ética es la disciplina que trata de la apreciación moral de los actos humanos, además de ser un conjunto de principios y de normas morales que regulan las actividades humanas.

La ética provee los saberes teóricos y prácticos para poder emplear bien la libertad. De acuerdo a Quispe-Otazu (2007) "Nadie es capaz de vivir bien con sólo desearlo". Hace falta tener claro en qué consiste vivir bien y después ponerlo en práctica. Las buenas intenciones no bastan.

La ética no estriba en gustos, opiniones o apetencias. No da lo mismo comportarse de un modo u otro. Es una parte importante de la vida de un hombre. Por ello, la ética toma parte en los diversos problemas que afectan a cada una de las profesiones humanas.

De ahí la gran importancia de definir reglas de comportamiento que guíen el correcto obrar.

Por otro lado la informática es actualmente una de las áreas de mayor influencia cultural a tal grado que ha logrado en cierta medida influir en nuestras conductas y en el modo de sentirnos parte de la sociedad.

En este orden de ideas y en contexto con el tema podemos manifestar que parte de la problemática ligada a las tecnologías de información y comunicación podemos citar a los:

- Fraudes por internet.
- La problemática del software y la propiedad intelectual: copias ilegales, calidad del software, fallas de software.
- Almacenamiento de datos y la invasión de la intimidad.

La piratería, mejor definida en el contexto en el que estamos trabajando como la venta y distribución ilegal de copias de software; es un mal que aqueja severamente la economía mundial. Desafortunadamente este fenómeno se encuentra muy arraigado en economías en desarrollo, donde no existen los factores culturales, legislativos o económicos que lo prevengan. De acuerdo a la Alianza de Software de Negocios (BSA por sus siglas en inglés), el promedio mundial de software pirata instalado en 2003 fue de 36%, representando pérdidas de aproximadamente 29 mil millones de dólares para la industria de tecnologías de información. Esta tasa es aún mayor en las economías en desarrollo (Rodríguez A. G., Vidrio Barón S. B. 2005).

Normalmente los esfuerzos en torno al combate a la piratería se centran en la formación de leyes robustas para protección de trabajo intelectual, así como en la elaboración e implantación de mecanismos para asegurar su cumplimiento. Adicionalmente los organismos más activos en la lucha contra la piratería, como la BSA, se forman por gremios de las mismas compañías desarrolladoras de software comercial, por lo que no consideran la promoción de software libre como una opción viable. En ocasiones, algunos esfuerzos mercadológicos se centran en la promoción de valores éticos, pero esto contrariamente a lo esperado da resultados inversos (Ibíd.).

Puede, en cierto sentido, argumentarse que las razones principales de la piratería son aquellas relacionadas con los factores socio-económicos de las regiones. La falta de capacidad de compra, teniendo además la necesidad de permanecer competitivas, impulsa



a las personas a modificar sus estándares morales para justificar una práctica que es incorrecta. Las campañas contra la piratería tienen poca efectividad por su carácter punitivo, en vez de preventivo. Es importante notar que para hacer una campaña efectiva deben atacarse las causas y no las consecuencias del problema. Así, en un enfoque más preventivo y adecuado, los esfuerzos deberían concentrarse en generar toda una cultura organizacional en las escuelas, principalmente en las que nos atañen a nosotros como docentes de ingeniería. Generar esquemas que permitan hacer más accesibles las aplicaciones a los consumidores finales, al igual que en el reforzamiento de valores, que tanto adolece nuestro país. Y entre esos valores, podemos ofrecer alternativas capaces de hacer que desde los primeros años y desde las primeras asignaturas se fomente el uso de software de libre adquisición y distribución, buscando con ello también de manera indirecta, que las empresas que venden software “propietario” bajen sus precios al público en general, al existir una competencia leal por supuesto y con ellos provoquen tales hechos; en un mercado que seguirá dominado por la ley de la oferta y la demanda.

En algunas ocasiones la falta de indicaciones por parte de las coordinaciones de asignaturas en algunas escuelas, produce que los docentes no modifiquen su actuar respecto a las herramientas informáticas que han venido utilizando desde que ellos se han graduado. Por lo que en ocasiones ya “casados” o decantados por una marca comercial de software en particular, difícilmente optarán por alguna de libre distribución o adquisición, a menos que se lleve a cabo una reforma que por ley modifique éste actuar, haciendo o promocionando software no digamos ya de libre adquisición, pero sí un software al alcance de todos los alumnos. Incluyendo el uso de software de libre adquisición en los planes y programas de estudio ayudará mucho a este fin.

Por otro lado, deberá dotarse de los recursos necesarios a los centros de investigación, pero más aún a los centros de desarrollo tecnológico que dentro de sus propósitos estén el desarrollo de software de libre distribución, ya que de esta manera se resolverá en cierta medida el problema de las copias ilegales y también cada nación que impulse estas iniciativas dejará en cierta forma de ser dependiente de otras; en cuanto a sus desarrollos informáticos y al mismo tiempo dejará de ser señalado como un país en donde la corrupción e impunidad en estos delitos imperan.

De igual forma, en la creación y/o en la renovación de los planes y programas de estudio en ingeniería, deberá evitarse en la

medida de lo posible la inclusión de asignaturas hechas exprofeso para utilizar un determinado software propietario, lo cual sucedía en el pasado, ya que la existencia de paquetes informáticos de libre distribución era nulo e incluso la variedad del software propietario para algún tópico específico en la práctica de la ingeniería era muy limitado. Sin embargo actualmente, salvo por alguna actividad muy específica, existen variadas alternativas libres y comerciales para la enseñanza de diversos tópicos de la ingeniería.

Por tales hechos y para ayudar a encontrar alternativas para la enseñanza de la ingeniería utilizando paquetes computacionales al alcance de todos, en el siguiente segmento se presentarán software de libre distribución, con una pequeña descripción de su origen y de su uso de acuerdo a sus propósitos específicos.

## Algunas alternativas

Software de adquisición y distribución libre en la enseñanza de la Ingeniería.

### wxMaxima

De acuerdo a Rodríguez Galván (2007), Maxima es un programa cuyo objeto es la realización de cálculos matemáticos, tanto simbólicos como numéricos; es capaz de manipular expresiones algebraicas y matriciales, derivar e integrar funciones, realizar diversos tipos de gráficos etc. Su nombre original fue Macsyma (MAC's SYmbolic MANipulation System), donde MAC, Machine Aided Cognition, era el nombre del Laboratory for Computer Science del Instituto Tecnológico de Massachusetts (MIT). Durante la fase inicial del proyecto, Macsyma se desarrolló en estos laboratorios a partir del año 1969 con fondos aportados por varias agencias gubernamentales norteamericanas (National Aeronautics and Space Administration, Office of Naval Research, U.S. Department of Energy y U.S. Air Force). El concepto y la organización interna del programa están basados en la tesis doctoral que Joel Moses elaboró en el MIT sobre integración simbólica. Según Marvin Minsky, director de esta tesis, Macsyma pretendía automatizar las manipulaciones simbólicas que realizaban los matemáticos, a fin de entender la capacidad de las computadoras para actuar de forma inteligente. El año 1982 es clave. El MIT transfiere una copia de Macsyma a la empresa Symbolics Inc. para su explotación económica,

haciendo el código propietario y otra al Departamento de Energía, esta copia será conocida con el nombre de DOE-Macsyma. En 1992 la versión comercial de Macsyma sería adquirida por una empresa que se llamaría precisamente Macsyma Inc, y el programa iría perdiendo fuelle progresivamente ante la presencia en el mercado de otros programas similares como Maple o Mathematica, ambos inspirados en sus orígenes por el propio Macsyma. Pero ocurrieron dos historias paralelas. Desde el año 1982, y hasta su fallecimiento en el 2001, William Schelter de la Universidad de Texas mantuvo una versión de este programa adaptada al estándar Common Lisp en base a DOE-Macsyma, la cual ya se conoce con el nombre de Maxima para diferenciarla de la versión comercial. Actualmente, el proyecto está siendo mantenido por un grupo de desarrolladores originarios de varios países, asistidos y ayudados por otras muchas personas interesadas en Maxima. Puesto que Máxima se distribuye bajo la licencia GNU-GPL, tanto el código fuente como los manuales son de libre acceso a través de la página web del proyecto <http://maxima.sourceforge.net>

## Octave, Python y Rubi

Un script o guión es una serie de órdenes que se pasan a un intérprete para que las ejecute. No cumplen la definición de programa porque no son ejecutables por ellos mismos. Un programa se comunica directamente con el sistema operativo mientras que un script lo hace con un intérprete que a su vez envía comandos al sistema operativo. Los lenguajes de scripting más conocidos son, en el caso de los lenguajes de uso general, Java, Python y Ruby. La popularidad de Java se debe a su naturaleza de producto comercial muy sencillo de administrar mientras que Python y Ruby son Software Libre; de igual o más calidad pero sin publicidad. Python es un lenguaje basado en la consistencia que ofrece una gran productividad y versatilidad. Ruby es uno de los lenguajes más recientes, su popularidad está aumentando gracias a la aplicación Ruby on Rails orientada al desarrollo de páginas web. Existe una gran variedad en los lenguajes de scripting orientado a matemáticas. Matlab, Maple, Mathematica, Scilab, Octave, Euler, O-Matrix, R o S son lenguajes de scripting. Los más conocidos son Matlab, Mathematica y Maple. (Hernández A. D. A. 2007).

## Psim

De acuerdo a Bargalló P. R.; Morón R. J. de Sust i Rossello J. (2009) PSIM es una herramienta de simulación de circuitos eléctricos y electrónicos por computadora. La programación resulta muy sencilla e intuitiva, ya que se hace por medio de una interfaz gráfica

que permite dibujar los esquemas de los circuitos que se desea simular, disponiendo para ello de barras de elementos que incluyen además de los generadores y las cargas, todos los elementos de control necesarios (aparatos de medida, control y estimación etc.). La empresa que comercializa Psim es Powersim, <http://www.powersimtech.com/>, de donde se puede descargar una versión de forma gratuita, aunque limitada en sus capacidades. Powersim se dedica al desarrollo de herramientas para simulación y diseño de productos como fuentes de alimentación, controladores de motores, conversión de potencia y sistemas de control. Su idea es incrementar la eficiencia del cliente y su productividad para reducir el coste y el tiempo de salida al mercado de nuevos productos.

## FEMM

Finite Element Method Magnetics (FEMM), es una herramienta de software libre muy versátil. El cual puede realizar las siguientes tareas:

- Análisis 2D (3D para sistemas axisimétricos)
- Análisis magnetoestático, corrientes inducidas, electrostático, conducción DC y AC.
- Análisis térmico en régimen permanente y transitorio.
- Incluye los efectos de saturación y laminado del material.
- Trabajo con imanes permanentes.
- Sin limitación del número de elementos.
- Trabajo por lotes. Lenguaje de programación propio (LUA)
- Post-proceso muy potente con cálculo de magnitudes de interés (par, impedancia equivalente, pérdidas por efecto Joule y magnéticas, etc.).
- Disponibilidad de un foro de discusión en Internet tanto para solucionar problemas como para proponer ampliaciones del programa.
- Enlace con otros programas: MATLAB/OCTAVE, EXCEL, OptiY (Ibíd.)

## OptiY

Este software cuenta con una versión disponible gratuitamente, la cual está restringida en cuanto al número de variables a optimizar, es ideal para los estudiantes de ciencias, ingeniería y economía y para investigadores en ciencias. Es un programa multidisciplinario de análisis y optimización que incorpora las técnicas más modernas y las estrategias de optimización más avanzadas. Presenta como ventaja el que los modelos de optimización son considerados como una caja negra con entradas y salidas. Permite enlazarse con paquetes de simulación como MATLAB, FEMM, ANSYS y otros. Permite realizar un análisis de sensibilidad, traza de curvas y mapas de soluciones y de minería de datos. La adaptación a un entorno especial de simulación se lleva a cabo por las interfaces adecuadas.

Ayuda a automatizar parte de la solución de los problemas de ingeniería y ciencias, en el proceso de diseño y desarrollo. La esencial por lo tanto, es la transformación de un problema en una serie de tareas de análisis y optimización. Las versiones de OptiY libre pueden ser obtenidas en <http://www.optiy.de>.

## PowerWorld Simulator

En su versión demostrativa que es para uso educativo únicamente y de distribución gratuita en la WEB, se pueden realizar todos los estudios disponibles con la versión original pero en sistemas de hasta 12 buses o nodos. Esto resulta suficiente para fines de aprendizaje y de comprensión en la simulación de sistemas de energía eléctrica para el caso de estudios de flujos de potencias, fallas por cortocircuito y demás estudios afines. Incluso permite visualizar el intercambio de potencia entre dos áreas de un sistema interconectado. Esta versión puede ser obtenida en la dirección URL de PowerWorld Corporation (<http://www.powerworld.com>).

## PSCAD

El PSCAD son las siglas de Power System CAD y significa Diseño Asistido por Computador de Sistemas de Potencia. Esta herramienta permite, a partir de la introducción de un esquema eléctrico, simular su comportamiento y analizar los resultados, todo ello en un entorno gráfico de manejo sencillo e intuitivo. Tiene, en consecuencia, herramientas integradas de representación de variables, medidores, elementos de control y modelos de componentes eléctricos. Uno de

los puntos fuertes del PSCAD es su biblioteca de componentes, que la utilización directa en un esquema eléctrico de los componentes más habituales en los sistemas eléctricos. Este programa está accesible de forma gratuita en su versión de estudiante con prestaciones limitadas, así como sus manuales y ejemplos de aplicación, desde la página web: <http://www.pscad.com>.

## Scilab

Scilab es un programa que se elaboró para la solución de sistemas de control, procesamiento de señales y otras aplicaciones matemáticas, con la filosofía del software libre y amparado con licencia GPL. Scilab fue creado para hacer cálculos numéricos aunque también ofrece la posibilidad de hacer algunos cálculos simbólicos como derivadas de funciones polinomiales y racionales. Posee cientos de funciones matemáticas y la posibilidad de integrar programas en los lenguajes más usados (Fortran, Java, C y C++). La integración puede ser de dos formas: por ejemplo, un programa en Fortran que utilice Scilab o viceversa. Scilab fue hecho para ser un sistema abierto donde el usuario pueda definir nuevos tipos de datos y operaciones entre los mismos. Scilab viene con numerosas herramientas: gráficos 2-D y 3-D, animación, álgebra lineal, matrices dispersas, polinomios y funciones racionales, Simulación: programas de resolución de sistemas de ecuaciones diferenciales (explícitas e implícitas), Xcos: simulador por diagramas en bloque de sistemas dinámicos híbridos, Control clásico, robusto, optimización LMI, Optimización diferenciable y no diferenciable, Tratamiento de señales, Grafos y redes, Scilab paralelo empleando PVM, Estadísticas, Creación de GUIs, Interfaz con el cálculo simbólico (Maple, MuPAD), Interfaz con TCL/TK. Este programa está disponible en <https://www.scilab.org/>.

## Geogebra

Es un sistema de geometría dinámica. Se pueden realizar construcciones de puntos, vectores, segmentos, rectas y secciones cónicas como funciones y, después se pueden cambiar dinámicamente. Es un programa matemático elaborado por M. Hohenwarter y un equipo internacional de desarrolladores en la Universidad de Salzburgo y la Universidad de Atlantic, Florida. El programa engloba geometría, álgebra y cálculo y está pensado para ser utilizado en todos los niveles educativos. Es un programa libre desarrollado bajo una licencia Creative Commons; puedes copiar, distribuir y transmitir el programa con finalidades no comerciales. Se trata de un programa de calidad que ha recibido varios premios

internacionales, como por ejemplo el European Academic Software Award, en el año 2002, el International Free Software Award, en el 2005, o el Tech Awards de Silicon Valley, en el 2009.

## GanttProject

Es una herramienta muy completa a la hora de planificar un proyecto, ya que permite una visualización absoluta del mismo. Todo queda bajo el control del programa, desde los recursos necesarios en materia personal, así como los días festivos, hasta dividir el proyecto en un árbol de tareas y asignar a cada uno los recursos oportunos. GanttProject permite establecer dependencias que interrelacionan las tareas, es decir, una tarea no podrá comenzar hasta que no finalice la anterior. Permite exportar la tarea realizada a una imagen JPG, PNG, PDF y HTML. El programa puede instalarse para ser empleado desde sistemas Windows, Linux, Mac u otros, en los que esté disponible una versión del entorno de ejecución Java (lenguaje multiplataforma sobre el que está desarrollado). El programa se encuentra bajo una licencia GNU que, básicamente, indica que el programa puede ejecutarse, copiarse, modificarse y distribuirse de forma gratuita por todos sus usuarios, pero manteniendo el tipo de licencia en sí, es decir, que sus copias o modificaciones no pueden distribuirse prohibiendo su reproducción. Puede descargarse en <http://prdownloads.sourceforge.net/ganttproject/ganttproject-2.0.2.exe?downloadp>.

## Conclusiones

El poco o nulo conocimiento de la existencia de software de libre distribución y adquisición de parte de algunos docentes de ingeniería, aunado a las recomendaciones que estos hacen de utilizar software comercial para los proyectos, tareas o deberes, provoca en los estudiantes la adquisición de copias ilegales, y con ello toda una avalancha de problemas posteriores que incluye el crecimiento de una actividad comercial ilegal debido a la demanda de tales productos en el mercado negro, mismo que puede evitarse ofreciendo alternativas, como las que se detallan en este trabajo.

Conscientes de la actitud ilícita que podríamos provocar en los alumnos, los profesores, no podemos ser guardianes de los intereses de una empresa y tampoco promotores de prácticas indebidas,



permitiendo que los alumnos realicen actividades ilegales, como la copia de programas informáticos cuyos autores no lo permiten, tan sólo por entregar unas tareas. En el otro extremo se sitúa el caso del software de libre distribución y adquisición el cual por antonomasia permite que sea instalado en tantas computadoras como sea conveniente y que los profesores compartan con sus alumnos, con todas las de la ley, las herramientas utilizadas (quizás, acompañadas de material docente propio), facilitándoles reproducir en sus hogares el entorno de trabajo del aula. Más aún, al usar en el aula una herramienta con licencia libre, los profesores cuentan con ventajas adicionales a la hora de la planificación y el desarrollo de la asignatura, derivadas de tener la garantía de que los programas podrán ser instalados y usados por los alumnos en su propio domicilio y además podrán ser instalados y usados por el profesor en tantos puestos como sea necesario.

El software con licencia libre permite y de hecho fomenta el disponer de varias herramientas a la vez, complementarias o capaces de interactuar entre sí, cada una de las cuales contará con sus puntos fuertes y sus debilidades. Aunque el profesor se decante por una de ellas, siempre podrá ofrecer a sus alumnos la enriquecedora posibilidad de experimentar con otras, de resolver un mismo problema desde distintas perspectivas y de saciar su curiosidad a aquellos que cuenten con mayores inquietudes. De igual forma ésta forma de conducirse (actitud) permeará por razones naturales entre el estudiantado, por lo que es menester que ésta buena práctica se fomente desde la llegada de las nuevas generaciones, (desde el primer semestre), para que a través de los profesores, en la práctica docente exista una verdadera y auténtica renovación moral, que tan necesaria es en la realidad cotidiana y no sólo en los discursos.

Ayudaría mucho también, como se mencionó en la investigación, que en los planes y programas de estudio de ingeniería no se creen asignaturas hechas ex profeso para utilizar algún software comercial en específico, salvo que ello otorgue ventajas competitivas tanto a los alumnos como a la universidad en cuestión, ya que las más de las veces en vez de beneficiar a la escuela se crea una dependencia tecnológica que pudiera evitarse desarrollando alternativas que incluso podrían surgir de las mismas tareas de investigación en las universidades, buscando con ello también un mayor y mejor desarrollo tecnológico en las naciones que no lo tienen.

Actualmente en Economías de primer mundo, como: Alemania, Reino Unido, España Francia, existen programas que motivan tanto



la investigación como el desarrollo de software de libre adquisición y distribución tanto para el uso de sus portales de internet, como para montar páginas e impartir la docencia en las principales universidades de estas naciones, en el caso de América Latina, tanto Brasil, como Colombia y Venezuela están apostando fuertemente hacia la utilización de software libre.



## Bibliografía

1. *Nuevo Modelo Educativo*. Secretaría Académica. IPN. México. 2002, recuperado en julio de 2003, disponible en [www.ipn.mx/SiteCollectionDocuments/RYSDoctentes/PUBLICACI\\_\\_N\\_I8437.pdf](http://www.ipn.mx/SiteCollectionDocuments/RYSDoctentes/PUBLICACI__N_I8437.pdf)
2. **Rodríguez Riotorto M.** (2011) *Primeros pasos en Maxima*. Recuperado en abril de 2013, disponible en <http://riotorto.users.sourceforge.net>.
3. **Rodríguez Riotorto M.** (2012) *Manual de Maxima*, versión 5.28. Recuperado en mayo de 2013, disponible en: <http://maxima.sourceforge.net/docs/manual/es/maxima.html>
4. **Hernández A D. A.** (2007) *Unidad de Diseminación de Software Libre*. Recuperado en abril de 2013 disponible en <http://nux.ula.ve> Corporación Parque Tecnológico de Mérida Universidad de Los Andes Introducción a GNU Octave Versión 1.0 – Agosto 2007.
5. **Rodríguez Galván J. R.** (2007) *Maxima con wxMaxima: software libre en el aula de matemáticas*. Departamento de Matemáticas. Oficina de Software Libre de la Universidad de Cádiz Febrero de 2007, recuperado en febrero de 2013, disponible en [https://forja.rediris.es/docman/view.php/2/3/guia\\_wxmaxima.pdf](https://forja.rediris.es/docman/view.php/2/3/guia_wxmaxima.pdf)
6. **Stallman. R. M.** (2004) *Software libre para una sociedad libre*. Recuperado en enero de 2013 disponible en [http://www.gnu.org/philosophy/fsfs/free\\_software.es.pdf](http://www.gnu.org/philosophy/fsfs/free_software.es.pdf)
7. **Quispe-Otazu, R.** (2007) *Ética, Informática y Sociedad*. Blog de Rodolfo Quispe-Otazu [Internet]. Noviembre 2007. Recuperado el 30 de Abril del 2012, Disponible en: <http://www.rodolfoquispe.org/blog/etica-informatica-y-sociedad.php>
8. **Bargalló P. R.; Morón R. J. de Sust i Rossello J.** "Herramientas informáticas de libre distribución y su aplicación al análisis y diseño de las máquinas eléctricas: aplicación al análisis del motor asíncrono trifásico mediante el método de los elementos finitos". EUETIB UPC Recull de contribucions congressos, cursos i seminaris convocatòria 2009-2010 SOTSDIRECCIÓ RECERCA recuperado en noviembre de 2011 y disponible en: [www.euetib.upc.edu/la-recerca/informes-dactivitat/.../file](http://www.euetib.upc.edu/la-recerca/informes-dactivitat/.../file)
9. **Rodríguez Abitia G., Vidrio Barón S. B.** *El Dilema de la Piratería de Software: Cuestión de Ética*. ITESM Campus Estado de México y la Universidad de Colima. México recuperado en julio de 2013 y disponible en <http://docente.ucol.mx/juancont/documentos/cap02/4.pdf>
10. **Bayón L., Grau J.M., Otero J.A., Ruiz M.M., Suárez P.M.** *Uso de herramientas de Software Libre para la enseñanza de las Matemáticas en los nuevos Grados*. Departamento de Matemáticas, EPI Gijón, Universidad de Oviedo. Presentado y recuperado de las memorias del XIX CUIEET Congreso Universitario de Innovación Educativa en Enseñanzas Técnicas, julio 2011, Barcelona, Catalunya.

