

# **EL SOFTWARE LIBRE: FUENTE DE RECURSOS EDUCATIVOS**

## **OPEN SOURCE (SOFTWARE LIBRE): SOURCE OF EDUCATIONAL RESOURCES**

M<sup>a</sup> del Mar Sánchez Vera<sup>1</sup> (mallemar@hotmail.com)

*Universidad de Murcia*

### **PALABRAS CLAVE:**

Software libre, recursos, TIC

### **WORDS FIX:**

Open Source, resources, TIC

### **RESUMEN**

Desde hace siglos, el hombre ha reflexionado acerca de la libertad, sus características y sus consecuencias. El famoso político estadounidense Abraham Lincoln decía que el hombre nunca ha encontrado una definición para la palabra libertad; sin embargo, suele considerarse que la libertad es lo que permite al hombre decidir si quiere hacer algo o no, y esta decisión lo hace libre, pero también responsable de sus actos. El software libre, como movimiento social a favor de la libertad, lucha contra el monopolio informático y promulga una libertad de acceso, de modificación y de intercambio de software, considerado como un bien que debería estar al beneficio de todas las personas, como causa justa para que los ciudadanos nos desenvolvamos en la sociedad de la información. De esta manera se plantea la posibilidad de que un hombre decida si quiere modificar o no un programa, y se asegure de que tenga la libertad para hacerlo. Pero, fuera de toda causa social o política, el software libre ofrece a la educación, una infinita gama de posibilidades de acceder recursos, recursos que además pueden ser modificados e intercambiados de modo que se creen verdaderas redes de colaboración entre los profesionales de la enseñanza.

### **ABSTRACT:**

For centuries, the man has thought brings over of the freedom, his characteristics and his consequences, the famous American politician Abraham Lincoln was saying that the man has never found a definition for the word liberate, Nevertheless, is in the habit of being thought that the freedom is what allows to the man to decide if ihe wants to do something or not, makes it free, but also person in charge for his acts. The open source (or software libre), as social movement in favour of the freedom, struggle against the computer monopoly and it promulgates a freedom of access, of modification and of interchange of software, considered as a good that should be to the benefit of all the persons, as just reason in order that the citizens we develop ourselves in the society of the information. So that a man decides if he wants to modify or not a program, and he makes sure himself that should have the freedom to do them. But, out of any social or political reason, open source (software libre) offers to the education, an infinite scale of possibilities of acceding resources, resources that besides can be modified and interchanged so that they believe real nets of collaboration the professionals of the education.

---

<sup>1</sup> Facultad de Educación de la Universidad de Murcia, Campus de Espinardo, apartado 4021, C.P. 30100, Murcia.

## **1. NUESTRO ENTORNO ACTUAL: LA SOCIEDAD DE LA INFORMACIÓN**

Desde hace miles de años, el hombre ha aprovechado la capacidad de modificar la naturaleza en un grado u otro para beneficiarse en algún aspecto. Ortega y Gasset, en 1970, afirmó que “el hombre, merced a su don técnico, hace que se encuentre siempre en su derredor lo que ha menester—crea, pues, una circunstancia nueva más favorable, segrega, por decirlo así, una sobrenaturaleza, adaptando la naturaleza a sus necesidades”. Ortega y Gasset consideraba, por tanto, la Técnica como transformación, como adaptación de la naturaleza a las necesidades del sujeto, nunca como adaptación del sujeto a los requerimientos de la naturaleza. Esta propiedad humana que Ortega y Gasset llamó sobrenaturaleza pone de manifiesto que las tecnologías forman parte de la historia de la humanidad, de la misma forma que la naturaleza.

De acuerdo con Selles, Pérez y Boronat (2006) el desarrollo tecnológico se entiende como una evolución ascendente, donde cada peldaño conduce necesariamente al siguiente. Fue durante los sesenta cuando se comenzó a hablar de la “Sociedad de la Información”, con la idea de que la información es poder, y pasando de esta forma a convertirse en un bien de consumo. Sin embargo, la posesión de información no es equivalente a la posesión de conocimiento, para que la información sea conocimiento son necesarias una serie de estrategias de análisis y profundización de la misma. (Cisneros, García y Lozano, 1999). En cualquier caso, el desarrollo de la telemática y la presencia de las Nuevas Tecnologías han transformado las diversas variables que configuran nuestra vida profesional y social.

Siguiendo la definición de González, Gisbert et al, en 1996 se entiende las Nuevas Tecnologías de la Información y la comunicación como “el conjunto de procesos y productos derivados de las nuevas herramientas (hardware y software), soportes de la información y canales de comunicación relacionados con el almacenamiento, procesamiento y transmisión digitalizados de la información”. Al hablar de las TIC se tienen en cuenta una serie de características (Martínez, 1994, Solano, 2004, Prendes, 1999):

- Sistemas de comunicación
- Informática
- Herramientas ofimáticas que contribuyen a la comunicación

Todas son importantes, pero es el hecho de posibilitar nuevos canales comunicativos lo que ha dado la posibilidad de transformación social que se ha producido entre las personas.

## **2. LAS TIC EN LA EDUCACIÓN**

El rápido desarrollo de las Tecnologías de la Información, han provocado continuos cambios en todos los aspectos de la vida colectiva y personal, dando la posibilidad de crear y difundir redes de conocimiento.

En general, las escuelas, encuentran frecuentes barreras a la hora de acoger las innovaciones tecnológicas (Martínez y Prendes, 2003), y en muchos casos, nos encontramos con un modelo de escuela tradicional opuesto al nuevo modelo de sociedad digital que se está promoviendo con el desarrollo de la tecnología y su aceptación en las tareas cotidianas. La exigencia de un reajuste curricular en todos los niveles educativos, ha impulsado el desarrollo de variadas experiencias educativas, que introducían el uso de las TIC en el aula como algo necesario para el aprendizaje de unos alumnos que, en la vida laboral y social, se tendrán que enfrentar a estas tecnologías. Si hasta ahora la escuela cumplía una función de almacenamiento de la información (Cabero, 2001), las TIC superan esa tarea, al posibilitar al estudiante el acceso a innumerables recursos.

Al hablar de la introducción de las TIC en la enseñanza, tendemos a centrarnos exclusivamente en sus potencialidades para el aprendizaje, y hay que comprender también su

influencia política e ideológica, y sus repercusiones en la configuración de un modelo de sociedad (Cabero, 2001). De acuerdo con Salinas (2004) se ha de tender hacia un modelo de educación flexible que se abra a las diferentes necesidades y lugares de aprendizaje.

### 3. EL SOFTWARE EDUCATIVO

La Real Academia Española de la Lengua define el software como “Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora” (RAE, 2006), el software son las instrucciones responsables de que el hardware realice su tarea<sup>2</sup>, es, en definitiva, lo que permite poner en relación al ser humano y a la máquina (y también a las máquinas entre sí). El término programa puede referirse al código fuente (las instrucciones de programa legibles por el programador), sin embargo, programa, en informática, es sinónimo de software, aunque cuando se habla de un programa, se supone cierto grado de terminación, es decir, se da por sentado que están ya presentes todos los archivos y las instrucciones que son necesarias para la interpretación del programa. (Enciclopedia Virtual de Informática, 2006). Marqués (1997) considera la definición de software educativo, programas educativos y programas didácticos como sinónimos que designan a los programas de ordenador creados con la finalidad de ser utilizados, para facilitar los procesos de enseñanza y aprendizaje (medios didácticos). El mismo autor señala, que siguiendo su propia definición, más basada en el criterio de finalidad que de funcionalidad quedarían entonces descartados los programas empresariales que también existen en los centros educativos con funciones didácticas pero con fines lucrativos.

Fuera de aspectos meramente informáticos, el desarrollo del software ha sido espectacular, con el consiguiente impacto en la educación. La revolución que se ha producido en el software, sobre todo en el interfaz de usuario, ha posibilitado que se pueda “reducir la edad mínima necesaria para poder utilizar un programa posibilitando el diseño de aplicaciones educativas para niños de muy corta edad (3 años)” (Romero, 2006).

### 4. SOFTWARE LIBRE

La Asociación de Usuarios Españoles de GNU/Linux, define el software libre como “el software que permite libertad a los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software”.

Ampliando un poco más esta idea, podemos decir que el software libre debe posibilitar lo que se denominan “las cuatro libertades”:

- *Libertad 0*: Ejecutar el programa, para cualquier propósito;
- *Libertad 1*: Estudiar el funcionamiento del programa, y adaptarlo a sus necesidades;
- *Libertad 2*: Redistribuir copias si con ello consideras que puedes ayudar a otro usuario.
- *Libertad 3*: Mejorar el programa, y poner sus mejoras a disposición del público, y todo ello exclusivamente con el propósito de beneficiar a toda la comunidad de usuarios del software libre.

---

<sup>2</sup> La Enciclopedia Virtual de Informática (<http://www.terra.es/personal/lermon/esp/enciclo.htm>) expone que, como concepto general, el *software* puede dividirse en varias categorías basadas en el tipo de trabajo realizado. Las dos categorías primarias de *software* son los sistemas operativos (*software* del sistema), que controlan los trabajos del ordenador o computadora, y el *software* de aplicación, que dirige las distintas tareas para las que se utilizan los ordenadores

El software libre enfatiza el propósito de mejorar, para compartir el programa con el resto de usuarios de estas aplicaciones, y el de permitir el acceso libre a todos los usuarios, que a la misma vez pueden seguir mejorando y haciendo crecer el conocimiento en una comunidad que se construye conjuntamente.

## 5. POSIBILIDADES DEL SOFTWARE LIBRE EN LA EDUCACIÓN:

El software nos posibilita formar parte de auténticas comunidades virtuales de software libre donde podamos colaborar con otros usuarios en la construcción compartida del conocimiento acerca de estas aplicaciones. Siguiendo a Mérou (2006), en general (y más concretamente en la enseñanza), el software libre ofrece muchas posibilidades:

- *Es abierto*: lo que implica que si una persona quiere curiosear el trasfondo de un programa puede hacerlo y modificarlo para adaptarlo a sus necesidades, la importancia de que sea abierto para la educación se hace mas evidente conforme avanzamos en el sistema educativo hasta llegar a la Universidad, y en el ámbito de estudiantes de informática es una ventaja obvia.
- *Se puede modificar*: como se comentaba, esta ventaja se deriva claramente de la anterior, el hecho de que sea un programa abierto y exista la posibilidad de acceder al código fuente nos permite transformarlo y proponer diferentes alternativas del programa.
- *Tiene un precio inigualable*: se refiere al hecho de que el coste del software libre suele ser muy bajo o nulo. Si utilizamos este tipo de software en la clase, estamos enseñando a los alumnos a manejar programas que no les va a costar ningún esfuerzo económico (o casi ningún esfuerzo) conseguir. Si usamos programas propietarios, cuando nuestros alumnos quieran utilizarlo en casa o fuera del centro tendrán que comprar (gastándose una considerable suma de dinero) o como ocurre en algunas ocasiones, piratearlos, lo cual constituye un delito de la propiedad intelectual.
- *La producción del Software Libre no depende exclusivamente de las ventas*: el software libre crece, en muchas ocasiones, por el placer intelectual y el reconocimiento profesional de quienes lo elaboran, lo cual, como hemos dicho anteriormente, favorece el bajo coste de este tipo de programas.
- *Sigue el modelo científico*: el modelo de producción de software libre (que se analizará posteriormente), se considera análogo al de producción de la ciencia. “La ciencia avanza proponiendo soluciones abiertamente, exponiéndolas al análisis y a la crítica de la comunidad y mejorando progresivamente con las aportaciones de más calidad. Y éste es exactamente el modelo de crecimiento del Software Libre. Los productores exponen sus programas a la comunidad que los critica, adopta o mejora con otras aportaciones”. (Mérou, 2005).
- *Facilita la participación*: esta es ya una ventaja realizada en muchas ocasiones. En los años 60, cuando comenzó el desarrollo de la informática, todo el software era libre, los programadores (llamados hackers, no como equivalentes a delincuentes informáticos) intercambiaban sus ideas y ficheros. Cuando comenzó la privatización del software, los partidarios del software libre continuaron (y continúan) con esta filosofía de colaboración y crecimiento conjunto.
- *Se propaga*: por todo el mundo: en el sentido de que en casi todos los países existen personas o grupos que se interesan por este movimiento y participan, gracias a Internet, con otras personas.
- *No necesita vender versiones para subsistir ni es tanta la diferencia entre versiones*: es esta una de las ventajas más interesantes que presenta el software libre ante el software propietario. El sistema de producción del software privado, como mecanismo industrializado, tiende a sacar cada cierto tiempo versiones nuevas de sus productos para continuar siendo competitivos en el mercado, no es esto un aspecto negativo, si

no fuera por el hecho de que estas versiones dan lugar a problemas de incompatibilidad con otras herramientas o sistemas que nos llevan a seguir consumiendo un software propietario específico para poder desarrollar una tarea. El software libre en cambio, permite ser modificado para adaptarlo a situaciones concretas.

- *Linux ha sido un éxito:* “Linux es la denominación de un sistema operativo y el nombre de un núcleo” (Wikipedia, 2006). Es el proyecto más conocido de software libre. El padre de Linux es Linus Torvalds, el cual en 1991 (siendo un estudiante de 21 años) expuso la idea de desarrollar un sistema operativo libre, se inspiró en MINIX, un pequeño sistema Unix. El despegue a nivel mundial de la Internet y el abaratamiento de las conexiones ha permitido que Linux se desarrolle rápidamente, y hoy sea uno de los sistemas operativos con más éxito, y uno de los más adoptados en las instituciones. En nuestro país, la Junta de Extremadura fue la primera en introducir a gran escala el software libre para las escuelas.

El software libre, tiene pues, muchas posibilidades. Sus potencialidades se han visto reforzadas porque permiten formar parte de auténticas comunidades virtuales de software libre donde podemos colaborar con otros usuarios en la construcción compartida del conocimiento acerca de estas aplicaciones y de la filosofía que subyace al software libre.

Algunas personas se contraponen a la idea del software libre argumentando que sus alumnos no serán programadores y que no necesitarán el código fuente, esto es cierto, hay muchas personas que no programaran ni serán informáticos, pero, estableciendo un símil con la carretera, no podemos enseñarle a nuestros alumnos a utilizar sólo las autopistas de pago que existen para ir de un lugar a otro, es nuestro deber mostrarles también las autovías nacionales, a las que podrán acceder libremente, y que llegan al mismo destino que las autopistas de pago. De igual modo, si algún alumno quiere crear un camino para ir hasta su casa, tiene que tener la posibilidad de hacerlo y compartirlo con los demás. La idea de trasfondo que se quiere mostrar es que la actitud de la escuela ha de ser abierta para mostrar a los ciudadanos del futuro las posibilidades que se les ofrece, y como se ha comprobado, el software libre ofrece muchas alternativas.

Centrándonos ahora en el aspecto más didáctico al hablar del uso de un determinado tipo de software, Belkin y Heinz, (2006), hablan de dos modelos enfrentados en función del software que se utilice:

<b>Modelo emparentado con el Software Propietario</b>	<b>Modelo emparentado con el Software Libre</b>
<p>Se enseñan herramientas informáticas. Ejemplo 1: Se enseña FrontPage. Ejemplo 2: Se enseña a usar Microsoft Access.</p>	<p>Se enseñan conceptos fundamentales que sirven como base para utilizar herramientas informáticas. Ejemplo 1: Se enseña a usar lenguajes HTML. Ejemplo 2: Se enseña qué es una base de datos y como usarlas.</p>
<p>Depende de determinadas herramientas informáticas. Ejemplo: Se enseña a guardar un archivo en Word.</p>	<p>La enseñanza depende de los fundamentos y no tanto de las herramientas. Ejemplo: Se enseña a guardar un archivo. Ese concepto con ciertas variantes se</p>

	aplican a otras herramientas ofimáticas.
<p>Se apoya primordialmente basándose en el aprendizaje visual. Se desestima el aprendizaje analítico.</p> <p>Ejemplo: Se les enseña a los alumnos a recordar las barras de herramientas de los programas.</p>	<p>Se enseña de manera tal que lo visual sirva para acelerar los conceptos analíticos adquiridos.</p> <p>Ejemplo: Abrir un programa es ejecutar un comando, lo cual significa que uno no depende de si le han borrado el icono del escritorio o de los menús. Si se sabe lo que significa ejecutar un comando y sabemos como hacerlo, podremos adaptarnos a diferentes configuraciones.</p>
<p>Todo depende de una empresa proveedora de software. Consciente o inconscientemente se lleva a los alumnos a una disyuntiva que es la de o usar copias prohibidas de software o bien pagar licencias onerosas.</p>	<p>El software libre es accesible a todos. No dependemos de una empresa en particular.</p>
<p>Se trabaja primordialmente sobre la base de resolver problemas de productos.</p> <p>Ejemplo: Como resolver bugs (defectos) de Microsoft Outlook.</p>	<p>Se basa en plantear soluciones a nuevos desafíos.</p> <p>Ejemplo: Hace algunos años, pese a tener sistemas operativos libres, no se contaba con aplicaciones fáciles de usar para los usuarios sin conocimientos técnicos.</p> <p>OpenOffice.org es un ejemplo de como una comunidad de usuarios finales, desarrolladores y empresas pueden trabajar en conjunto para obtener una herramienta amigable y sólida.</p> <p>Las instituciones educativas deberían fomentar el trabajo colaborativo de manera que los alumnos puedan hacer aportes a la comunidad. Que pasen de tener un papel de consumidores pasivos a activos.</p>
<p>Se fomenta un modelo de <i>Yo gano, Ud. pierde</i></p>	<p>Se fomenta un modelo colaborativo de <i>Yo gano, Ud. gana.</i></p>
<p>El modelo de desarrollo está en ocultar, y obstaculizar el acceso al código fuente.</p>	<p>El modelo de desarrollo se basa en compartir el código fuente de los programas, esto permite el progreso del software a través de la cooperación comunitaria.</p>

Se hace creer que la historia principal pasa por una determinada empresa.	Se aplica el método científico a la informática.
---	--

**Tabla 1:** Modelos de enseñanza con software propietario y con software libre. (Belkin, Heinz, 2006)

Algunas de las ideas desarrolladas por estos autores han sido ya tratadas anteriormente cuando se exponían las ventajas del software libre en la enseñanza, pero resulta interesante observar como una buena introducción de las herramientas del software libre en la enseñanza puede fomentar otro tipo de método educativo, basado en el planteamiento de soluciones compartidas.

## 6. SOFTWARE LIBRE Y CONTENIDOS EDUCATIVOS:

Rocheffoucauld, un escritor Francés del siglo XV dijo que “la libre comunicación de los pensamientos y las opiniones es uno de los derechos más preciados por el hombre”, desde el movimiento del software libre se considera que el software son pensamientos, ideas, que han surgido de la colaboración entre un grupo de personas. Traspasando esta idea a la educación, existen corrientes que tratan de fomentar la creación de contenidos educativos basándose en esta idea de creación conjunta que promueve el software libre.

Siguiendo a Monge (2005), en un estudio realizado a profesores de secundaria, entre el software libre y los contenidos educativos existen ciertas similitudes y algunas diferencias. En primer lugar, la mayor parte de similitudes provienen del objeto de producción, es decir, tanto si hablamos de software como de contenidos, estamos hablando de bienes inmateriales, de ideas, información, conocimiento...

Además, en la actualidad, la importancia de los objetos de aprendizaje (unidad mínima de contenido educativo) y su modularidad es similar a los proyectos de software libre, que funcionan de un modo descentrado de producción basado en la modularidad.

Sin embargo, existen diferencias claras entre ambos modelos de producción:

- La comunidad: en el software libre el concepto de comunidad es importante, los creadores de software libre tienen en mente que su trabajo va a poder ayudar a toda la comunidad, mientras que en el caso de creación de contenidos, por ejemplo entre unos profesores de secundaria, no es habitual que compartan el contenido que crean con los demás docentes.
- Manejo de las herramientas tecnológicas: primero porque para los creadores de software libre, la facilidad en el manejo de las TIC es necesaria para forjar una comunidad virtual, lo que significa que estamos hablando de unas personas que tienen cierta facilidad, naturalidad y costumbre de manejar las TIC, sin embargo, los profesores no hacen un uso tan cotidiano o no poseen el dominio tecnológico de estas herramientas, la comunidad de los docentes es más irregular con distintos niveles de manejo.
- Estándares: en los proyectos de software libre, normalmente, la mayoría del trabajo está programado en el mismo lenguaje, en educación, por el contrario, no existe un estándar para producir contenidos de manera que estos sean reutilizables. El impulso que están recibiendo los repositorios de objetos de aprendizaje suponen un esfuerzo de compaginar los recursos educativos. Pero existen aún muchas dificultades para unificar criterios, tal y como afirma Polsani, (2003), hay muchas controversias a la hora de unificar una definición de *Learning Object* (Objeto de Aprendizaje).
- Voluntarios universitarios: Monge (2005) explica cómo en la mayoría de proyectos de software libre la iniciativa y el fomento desde las universidades ha sido importante, la mayoría de proyectos de este tipo han surgido desde universidades. Cuando se traslada

este fenómeno a los contenidos educativos se comprueba que el tema no tiene el mismo impulso.

- Masa crítica: “Inicialmente, las computadoras eran herramientas que servían para procesar datos, y los programadores se ayudaban entre sí compartiendo el código que escribían.” (Matías, 2004). El fenómeno del software libre está muy expandido en la actualidad, pero para que se pudiera trasladar a la educación sería necesario que adquiriera una filosofía parecida, es decir, que un grupo de profesores, de forma totalmente altruista, se lanzaran a acometer un proyecto de manera que sus creaciones fueran compartidas por el resto de la comunidad educativa. Se refiere, al cambio hacia una actitud generosa y colaborativa entre docentes.

Como se ha comprobado, el proceso de creación de software libre y el de contenidos educativos no es totalmente equiparable, sin embargo, es cierto, que las ventajas del software libre no se reducen a los beneficios que su uso no suponen en cuanto a gratuidad, libertad...y el resto de ventajas que antes se enumeraban, sino que también podemos detenernos en su filosofía y su modo de creación colaborativa de conocimiento, de manera que la educación pueda adquirir algunas de las estrategias que se utilizan para el desarrollo del software libre y aprovecharnos de la experiencia que ha tenido.

## 7. SOFTWARE LIBRE COMO FUENTE DE RECURSOS EDUCATIVOS

El software libre ha desarrollado multitud de programas educativos, a través del intercambio de ideas y experiencias existe gran cantidad de software que siendo libre (y en muchos casos gratuito) abre muchas posibilidades de uso para la educación.

Se va a exponer una selección de programas de software libre educativos, estamos hablando pues, de software didáctico; siguiendo en la línea de Marqués (1997), son los programas educativos comparten cinco características esenciales:

- Son materiales elaborados con una finalidad didáctica,
- Utilizan el ordenador como soporte en el que los alumnos realizan las actividades que ellos proponen,
- Son interactivos,
- Individualizan el trabajo de los estudiantes,
- son fáciles de usar.

### 7.1 Algunos programas educativos de software libre


- JCLIC: (<http://clic.xtec.es/es/jclic/>) (Gratuito, disponible para Windows, Linux y MAC OS) está formado por un conjunto de aplicaciones informáticas que sirven para realizar diversos tipos de actividades educativas: rompecabezas, asociaciones, ejercicios de texto, palabras cruzadas...Las actividades no se acostumbran a presentar solas, sino empaquetadas en proyectos. Un proyecto está formado por un conjunto de actividades y una o más secuencias, que indican el orden en qué se han de mostrar. El proyecto JClíc es una evolución del programa Clic 3.0, una herramienta para la creación de aplicaciones didácticas multimedia con más de 10 años de historia.



Dibujo 1: Logo de Jclíc

JClíc está formado por cuatro aplicaciones:

- *JClíc applet*: Un "applet" que permite incrustar las actividades JClíc en una página web.
- *JClíc placer*: Un programa independiente que una vez instalado permite realizar las actividades desde el disco duro del ordenador (o desde la red) sin que sea necesario estar conectado a Internet.

- *JClic autor*: La herramienta de autor que permite crear, editar y publicar las actividades de una manera más sencilla, visual e intuitiva.
  - *JClic reports*: Un módulo de recogida de datos y generación de informes sobre los resultados de las actividades hechas por los alumnos.
- TUX PAINT: (<http://www.tuxpaint.org/download/>) (Gratuito, disponible para Windows, Linux y MAC OS) Es un programa de dibujo libre diseñado para niños chicos (3 o más años). Presenta una interfaz simple y fácil de usar, divertidos efectos de sonido y una estimulante mascota que ayuda a guiar al niño mientras utiliza el programa. Posee diversas herramientas de dibujo: pintar, sellos, líneas, figuras, texto...
 
  - KTOUCH: (<http://ktouch.sourceforge.net/>) (Gratuito, disponible para Windows, Linux y MAC OS) KTouch es un programa para aprender mecanografiado, ofrece un modo fácil de aprender a escribir a máquina sobre un teclado rápida y correctamente. Cada dedo tiene su lugar sobre el teclado con unas claves asociadas para apretar.

Dibujo 2: Programa Tux Paint en modo “rellenar”

- TUX TYPING: (<http://tuxtype.sourceforge.net/>) (Gratuito, disponible para Windows, Linux y MAC OS) Es una herramienta que, como la anterior, permite aprender a teclear; la diferencia es que esta está realizada especialmente para los niños pequeños, posee diferentes niveles de dificultad y juegos para hacer más dinámico el aprendizaje.
- RECUERDA MAS: (<http://www.thebestmemory.com/>) (Gratuito, disponible para Windows) Recuerda Más es un programa de ayuda al estudio. Permite crear colecciones de fichas con preguntas y respuestas que incluyen texto e imágenes e incluye varias colecciones de muestra. Las fichas pueden incluir textos (cortos y largos) e imágenes. Pudiendo clasificarse y ordenarse por varios criterios.
- CUBE TEXT: (<http://www.vandenoever.info/software/cubetest/>) (Gratuito, disponible para Windows y Linux) CubeTest es un programa para practicar la visión espacial, dirigido a la escuela primaria. El usuario tiene que contestar diez preguntas tipo test. En cada una de ellas, CubeTest muestra un cubo y cuatro posibles vistas diferentes del mismo cubo (sólo una de ellas es correcta).
- MEMORIZE WORDS FLASHCAT SYSTEM: (<http://memorize-words.sourceforge.net/>) (Gratuito, disponible para Windows y Linux) Memorize Words es un programa de aprendizaje de vocabulario inglés basado en flashcards. El programa permite crear flashcards que incluyen imágenes, ejemplos de utilización de las palabras y pronunciación descargados de Internet. Las sesiones de aprendizaje se basan en el método Leitner de repetición espaciada<sup>3</sup>.

<sup>3</sup> El **repaso espaciado** (en inglés, *spaced repetition*) es una técnica de aprendizaje en la que se repasa a intervalos crecientes. Normalmente, en el repaso espaciado no se realiza una simple relectura, sino que pretende ser un tipo de aprendizaje activo, al obligarse al individuo a reaccionar a preguntas o estímulos. En este contexto, suelen utilizarse reglas mnemotécnicas. ([www.wikipedia.org](http://www.wikipedia.org))

- KEDUCA: <http://keduca.sourceforge.net/> keduca surgió en Julio de 2000 con la idea de poder ser un programa de evaluación, tanto para estudiantes como para centros educativos o profesores. Se inició en Linux con licencia GPL de la Fundación de Software Libre (FSF) y aún en estado beta fue desarrollándose hasta su desarrollo estable en Junio de 2001. En ese momento, a petición del proyecto KDE fue introducido en la base del proyecto de educación en la que se mantiene y en la que también existen otros programas bastante interesantes. Por diversos motivos el programa ha mantenido un desarrollo bastante desacelerado pero que por la disponibilidad y la intención de mejorarlo, en estos meses ha incrementado el número de opciones disponibles, de las que todavía surgirán más. En estos momentos, acabo de terminar la versión para Mac OS X, escrita en Cocoa. A partir de ahora, se podría decir que la versión para OS X irá por delante de la versión Linux y será el centro de desarrollo.

En un documento de Gómez, Neri y Parente (2005) se analizan las herramientas didácticas con software libre, los autores destacan las áreas de interés general que se pueden trabajar con algunas herramientas:

- Matemáticas:

Kbrunch.- es un programa para aprender fracciones, nivel primaria

Kig.- programa que permite construir figuras geométricas, nivel secundaria

Kmplot.- programa que permite graficar funciones, nivel secundario

- Ciencia:

Kalziun.- programa que muestra información sobre la tabla periódica, nivel secundaria

Kstars.- simulador de constelaciones y estrellas, nivel secundaria

- Idiomas:

Kverbos.- Permite ejercitar las tiempos verbales en español, nivel primaria.

KvocTrain.- Vocabulario en varios idiomas que incluye niveles de avance, nivel primaria

KhangMan.- Juego interactivo para utilizar vocabulario con más de 20 idiomas, nivel secundaria

Klatin, Kletter.- Auxilian en el aprendizaje de algunas sílabas en diferentes idiomas

- Otros:

Kgeography.- Programa para aprender geografía, nivel secundaria

Ktouch.- Permite mejorar las habilidades en el uso del teclado, nivel primaria

Kturtle.- Es un programa para aprender programación, basado en lenguaje logo, nivel primaria

- Herramientas de enseñanza:

Keduca.- Permite crear y revisar evaluaciones de una manera muy sencilla, docentes

## 8. Conclusión

El software libre constituye hoy en día una fuente de recursos importante para la educación, además, hay que pensar que estos recursos son creados por una gran cantidad de personas, que partiendo de la colaboración, comparten su conocimiento con el fin de crear herramientas que favorezcan a la sociedad. Es positivo que desde la educación nos interese por este tipo de software que promueve el libre intercambio de conocimiento, su bajo coste nos permitirá además que estemos formando a nuestros alumnos con herramientas que ellos van a poder adquirir sin un excesivo esfuerzo económico.

## 9. Bibliografía

- Abella, A., Sánchez, J. y Segovia, M.A. (2004). “Libro Blanco del software libre en España”. [www.campus-oei.org/salactsi/254-2004.pdf](http://www.campus-oei.org/salactsi/254-2004.pdf)  
[Consultado el 21 de junio de 2006]
- Adell, J. (2005): “Tendencias en educación en la sociedad de las tecnologías de la información”, en *EduTEC revista electrónica*, UIB.  
<http://www.uib.es/depart/gte/revelec7.html>  
[Consultado el 18 de Octubre de 2006]
- Belkin, S., Heinz, F. (2006). “Comparando modelos educativos: Software Libre y Software Privativo/Propietario”.  
[http://wiki.gleducar.org.ar/wiki/Comparando\\_modelos\\_educativos](http://wiki.gleducar.org.ar/wiki/Comparando_modelos_educativos)  
[Consultado el 24 de Enero de 2007]
- Bernal, J., Blanco, H., Clerencia, I. (2004). “Historia y cultura del software libre”. GLUZ. PULSAR. Asociación para la promoción e implantación del Software Libre en la Universidad de Zaragoza. <http://pulsar.unizar.es/gluz/manual-sl/p33.html>  
[Consultado el 12 de junio de 2006].
- Cabero, J. (2001). “Tecnología Educativa: diseño y utilización de medios en la enseñanza”. Barcelona. Paidós, papeles de comunicación.
- Cisneros, I., García, C. y Lozano, I.M. (1999). “¿Sociedad de la información – sociedad del conocimiento?”. La educación como mediadora, en *Congreso EduTEC 1999, Nuevas Tecnologías en la formación flexible y a distancia*. Sevilla  
<http://tecnologiaedu.us.es/edutec/paginas/43.html>  
[Consultado el 25 de Octubre de 2006]
- Fraser, A. (2005). “Los 10 mitos de Moodle”. Campaña HUGToB.  
[http://docs.moodle.org/es/Los\\_10\\_mitos\\_de\\_Moodle](http://docs.moodle.org/es/Los_10_mitos_de_Moodle)  
[Consultado el 28 de Diciembre de 2006]
- Gómez, J.A., Neri, A.S., Parente, M.A. (2005). “Herramientas didácticas con software libre”.
- González. A.P., Gisbert, M., Guillen, A., Jiménez, B. Lladó, F. y Rallo, R. (1996). “Las nuevas tecnologías en la educación”, en Salinas J. et. al. (1996), *redes de comunicación, redes de aprendizaje*. EDUTEC 95. Palma. Universitat de les Illes Balears.  
<http://www.uib.es/depart/gte/grurehidi.html>  
[Consultado el 2 de Febrero de 2005]
- González, J, Seoane, J., Robles, G. (2005). “Introducción al software libre”. Biblioteca Digital. UOC, Fundació per a la Universitat Oberta de Catalunya.
- Himanen, P. (2001). “La ética del hacker y el espíritu de la era de la información”. Barcelona: Destino.
- Heinz, F. (2006). “Software libre y educación”.  
[http://www.masternewmedia.org/es/2006/10/31/software\\_libre\\_y\\_educacion.htm](http://www.masternewmedia.org/es/2006/10/31/software_libre_y_educacion.htm)  
[Consultado el 29 de Octubre de 2006]
- Marqués, P. (1997). “El software educativo”. Universidad Autónoma de Barcelona  
[http://www.lmi.ub.es/te/any96/marques\\_software/](http://www.lmi.ub.es/te/any96/marques_software/)

[Consultado el 12 de Diciembre de 2006]

Matías, E. (2004). “Una introducción al software libre”, en *Pueblos*, nº 14, pp. 49-51.

[http://www.revistapueblos.org/article.php3?id\\_article=73](http://www.revistapueblos.org/article.php3?id_article=73)

[Consultado el 27 de Enero de 2007]

Merou, R. (2005). “Argumentos a favor del software libre en las aulas”. BULMA.

Bisoños Usuarios de GNU/Linux de Mallorca y Alrededores.

<http://bulma.net/body.phtml?nIdNoticia=1831>

[Consultado el 26 de junio de 2006]

Mérou (2006). “Ventajas del software libre en la enseñanza”, Antena de comunicación, BULMA

[http://www.coitt.es/antena/pdf/163/06b\\_Software\\_libre.pdf](http://www.coitt.es/antena/pdf/163/06b_Software_libre.pdf)

[Consultado el 26 de Enero de 2007]

Monge, S. (2005,). “¿Es aplicable el modelo de producción del software libre a contenidos educativos?”, en *RED. Revista de Educación a Distancia*, número monográfico II.

<http://www.um.es/ead/red/M2/>

[Consultado el 24 de Enero de 2007]

Ortega y Gasset, J. (1970). “Meditación de la técnica”, en *Obras Completas*, V. Revista del occidente, Madrid.

Polsani, P. R. (2003). *Use and abuse of Reusable Learning Objects*.

<http://jodi.ecs.soton.ac.uk/Articles/v03/i04/Polsani/>

[Consultado el 27 de Enero de 2007]

Salinas, J. (2004). ”Hacia un modelo educación flexible: elementos y reflexiones”, en Martínez, F. y Prendes, (2004) (coords), *Nuevas Tecnologías en Educación*. Madrid. Pearson.

Solano, I.M., Sanchez, M.M., Terry, J.M. (2006). “Posibilidades del software libre en la enseñanza: proyecto Linux en clase”. IX Congreso EDUTECH 2006: La educación en entornos virtuales: calidad y efectividad en el e-learning. En Actas Congreso EDUTECH 2006. Tarragona. EDUTECH. Edición electrónica en CD. ISBN: 84-690- 0126-4.