



# SOFTWARE EDUCATIVO Y HERRAMIENTAS LIBRES PARA LA EDUCACION A DISTANCIA

Calidad y Materiales educativos y Herramientas Tecnológicas en Educación a Distancia.

Ing. Ariel Adolfo Rodríguez Hernández  
Universidad Pedagógica y Tecnológica de Colombia. Colombia -  
[ariel.rodriguez@uptc.edu.co](mailto:ariel.rodriguez@uptc.edu.co)

## Resumen

Concientes de la necesidad que la educación tiene de incorporar las tecnologías de la información y la comunicación a los procesos de enseñanza-aprendizaje el grupo de investigación TICA: Tecnología, Investigación y ciencia aplicada desde hace varios años viene desarrollando proyectos de incorporación de TICs en la educación.

Uno de estos proyectos es el que hoy se presenta en este documento y cuyo propósito ha sido identificar soluciones de software educativo: sistemas de simulación, entornos educativos virtuales, software educativo, entre los más destacados. Pero con un valor agregado, que deben estar desarrollados bajo estándares de software libre que garanticen su uso, modificación y ejecución sin restricciones para el usuario final o la institución que desee utilizarlos.

Alrededor de este tema se inicio una investigación desde el año 2007, la cual inicio con el tema de simuladores desarrollados en software libre para física, al ir desarrollando este tema fueron identificándose toda clase de soluciones de software libre, haciendo con ello que de forma paralela se empezara a inventariar estas soluciones.

Hoy por hoy se han identificado cerca de 2500 soluciones de software libre entre simuladores, entornos de aprendizaje y otras soluciones con uso educativo.



29 de octubre al 9 de noviembre de 2008

---

Y viendo este gran volumen se diseñó un repositorio de simuladores en software libre el cual se ha denominado Simula Open Source. Este portal tiene como propósito referenciar todo desarrollo de software educativo libre.

**Palabras claves:** simulación, software educativo, software educativo libre, software libre, software de simulación.

## **SOFTWARE EDUCATIVO Y HERRAMIENTAS LIBRES**

### **Introducción**

El software libre ha penetrado hoy por hoy todas las áreas del conocimiento y del desarrollo de la sociedad, su organización y modelo de desarrollo le ha permitido posicionarse como una verdadera solución a las necesidades computacionales e informáticas del mundo actual.

Pero poco se sabe de sistemas de software libre para uso educativo, y se hace necesario estudiar la evolución y el estado de desarrollo del software educativo libre.

Su objetivo principal fue identificar y documentar el estado de desarrollo del tema en la sociedad actual y transformar este estudio en un repositorio o banco de recursos digitales inicialmente concebido para simuladores que almacene los resultados de este estudio y los publique en la Internet mediante un sitio Web. Y posteriormente incluye todos los tipos de software educativo.

Definido este propósito el proyecto dio inicio al estudio sobre el tema y a lo largo de este documento se exponen los resultados obtenidos, categorizando y analizando los resultados bajo ciertas variables.

Se presenta estudios por tipo de software identificado, por tipo de lenguaje en que se desarrollaron, por tipo de licencia con el que se distribuyen, por la plataforma de sistema operativo en que se encuentran disponibles y para el caso de la física que es el tema central por área de conocimiento dentro de la física.

Este estudio lleva al lector por proyectos de todo el mundo desde lo local, proyectos en Colombia y pasando por Estados Unidos, Canadá, Argentina, Perú, Europa (España, Portugal, Francia, Italia, Alemania, Inglaterra) Asia (Japón, China, Taiwán) entre los países con proyectos identificados. Se identificaron en total 50 proyectos de simuladores en software libre, con un total de 2097 simuladores para física. Y en materia de entornos de aprendizaje alrededor de 6, y otro tipo de software educativo al momento de este artículo se han identificado más de 15 proyectos.

Muchos de estos proyectos son liderados por profesores de las diferentes universidades, otros por comunidades de software patrocinadas por universidades. Y por la comunidad internacional de software libre.



## 1. Identificación y clasificación del software

Para llegar a los resultados que se presentan en esta experiencia investigativa se hizo necesario partir de un problema el cual se planteó entorno a los siguientes interrogantes:

1. Existe software (simuladores, sistemas operativos, software educativo, entornos de aprendizaje) para uso educativo en la enseñanza-aprendizaje de la física, desarrollados en software libre?

Esto dio inicio al trabajo de campo que consistió en identificar software de las características mencionadas y documentarlo con el objeto de incluirlos en un repositorio o banco de simuladores.

Se trabajó durante largos once meses haciendo búsquedas en Internet, enviando cartas a universidades y comunidades de software libre a fin de obtener fuentes de información sobre el tema.

Paralelo a este trabajo y vinculando a un joven investigador, tecnólogo en sistemas se inició el diseño y desarrollo del repositorio en línea, que como en el proyecto inicial abordaba solo simuladores se denominó Simula Open source<sup>1</sup>.

Para efectos de organización de la información se definió un sistema de categorización, el cual incluyó el tipo lenguaje en el que se desarrolló cada software, el tipo de licencia con el que se distribuye y para la física la categoría a la que pertenece según la temática. Estas variables se utilizaron para clasificar y parametrizar los resultados.

## 2. Modelos de Licencias para el Software Libre

Las licencias son el instrumento legal para proteger la propiedad del software, en el caso del software libre por sus propias características éstas adquieren formas o modelos específicos:

### a. GNU Public Licence (GPL)

Implica la publicación del código fuente, sin restricción de copia y distribución; los cambios deben ser autorizados y los trabajos derivados deben publicarse también bajo la modalidad GPL, es decir, los productos derivados heredan la licencia. Este modelo de licencia, como dijimos, fue creado por Richard Stallman en 1984 y hoy se considera la piedra fundadora del movimiento del Software Libre.

### b. LGPL (Less GPL)

---

<sup>1</sup> <http://www.ticagi.org> Web site del grupo de Investigación TICA.



29 de octubre al 9 de noviembre de 2008

Es una licencia que implica también la publicación del código fuente y la libertad de modificar el código; sin embargo, permite el uso comercial de productos OSS; es decir, con software liberado bajo este tipo de licencia se puede lucrar.

**c. Mozilla Public Licence (MPL)**

Es una licencia desarrollada por Netscape para su browser Mozilla, permite incorporar (combinar) software MPL a software comercial o “contaminado”.

**d. MIT y BSD (Berkeley Software Distribution) Licences**

Son licencias desarrolladas por el ámbito académico -MIT y Berkeley- permiten el derecho a copia y documentación sin cargo; se puede redistribuir y modificar sin necesidad de que el nuevo producto herede el tipo de licencia, es decir, se pueden derivar productos comerciales, sólo requiere mencionar a los autores.

**e. Public Domain**

Implica el abandono total de todos los derechos de copyrights, permiten combinarlo con software comercial sin restricciones.

Otros dos modelos de licencia que suelen confundirse con software libre son freeware y shareware.

**f. Freeware Licences**

Esta representa una modalidad que permite el uso sin restricciones ni costo de un programa pero no se libera el código fuente, por consiguiente el usuario no puede estudiarlo ni modificarlo.

**g. El shareware**

Es una modalidad con alcances similares al anterior pero por un período determinado luego del cual el usuario debe pagar una licencia por el uso del software.

Ambos modelos de licencia -freeware y shareware- son utilizados frecuentemente por los fabricantes de software propietario para poner a disposición de los usuarios sus productos a modo de prueba o para facilitar su venta. Ninguno contempla la liberación del código fuente.

### **3. Ventajas del uso de software libre**

Las ventajas derivadas de usar soluciones (sistemas operativos y programas) basadas en software libre son:



29 de octubre al 9 de noviembre de 2008

1. **Bajo costo.** Es la primera motivación para el uso del software libre ya el costo de adquisición del software puede ser gratis o de costo muy reducido.
2. **Independencia** total de cualquier sector privado o empresa. Esto supone no estar ligado a las condiciones de mercado impuestas por empresas de software que algunas veces ostentan situaciones de monopolio.
3. **Seguridad y privacidad.** Al disponer del código fuente, se conocerá el funcionamiento interno y se encontrarán y corregirán los posibles errores, fallos y agujeros de seguridad. Actualmente Linux es inmune ante la inmensa mayoría de virus informáticos que afecta casi exclusivamente a los sistemas Windows.
4. **Adaptabilidad.** Las modificaciones y correcciones de posibles errores se realizan de forma inmediata. De esta forma, las aplicaciones están en continua mejora y proceso de evolución.
5. **Calidad.** El software libre, al ser de dominio público, está siendo continuamente usado y depurado por un gran número de desarrolladores y usuarios del mismo, que añaden y demandan constantemente nuevas funcionalidades.
6. **Respecto a los estándares.** El uso de software libre y sistemas abiertos facilita la interoperabilidad entre distintas organizaciones.
7. **Predistribución.** Cualquier cambio y mejora que se introduzca en programas bajo licencia libre debe ser incluido en versiones posteriores y añadido al código fuente. Así el desarrollo tecnológico es continuo y dinámico.
8. **No hay restricción legal de uso.** No hay limitación en el número de licencias ni de copias dentro de la organización como ocurre con el software no libre donde se establece el pago en función de número de usuarios, tamaño de la organización, etc.
9. **Continuidad.** Se garantiza el derecho de cualquier usuario a continuar el desarrollo.
10. **Facilidad.** Se pueden iniciar nuevos proyectos basados en el código de un programa libre o adaptarlo sin necesidad de solicitar autorización al respecto.

#### 4. Fases de Desarrollo de la Investigación

##### 4.1. Fase de identificación y documentación de recursos de software

En esta fase se busco e identifiqué las soluciones de software educativo de tipo simulador para laboratorios de física desarrollados en software libre en el ámbito nacional e internacional. Esta etapa o fase incluyo que a cada solución encontrada y que halla sido desarrollada en los últimos diez años se le identificara sus características, funcionalidad, información técnica, el código fuente y los ejecutables en la medida de lo posible, o en su defecto el sitio Web desde el cual se puede utilizar.



29 de octubre al 9 de noviembre de 2008

Estos programas se seleccionaron previo análisis de las características que deben cumplir en términos del tema que abordan, el cual se desarrollo con base en los contenidos que se enseñan en la Física I de los programas de pregrado. Los cuales en particular deben dar tratamiento informático de la simulación de fenómenos relacionados con los temas.

Al mismo tiempo, en esta fase se hizo una revisión de trabajos sobre la problemática didáctica del aprendizaje experimental de la Física, con objeto de identificar los aspectos pedagógicos más importantes que debe cumplir un sistema de simulación para el uso educativo (contenidos teóricos, experiencias virtuales, tareas de desarrollo de destrezas, actividades sobre aprendizaje)

#### **4.2. Fase de diseño y desarrollo del repositorio de los recursos de software**

Previo al diseño y desarrollo se selecciono como herramienta de desarrollo del repositorio el PHP 5. como lenguaje base de programación para aquellas funciones adicionales que se requieran, y se utilizo para su diseño Web el entorno Aptana cuya licencia de uso es GPL. Como gestor de bases de datos MySQL, como servidor Web Apache. Para configurar el sitio web se hizo uso de cpanel.

En esta fase se hizo uso de la ingeniería de software para diseñar la solución Web que serviría como repositorio de los simuladores y su respectiva documentación. Diseño que incluyo la integración dentro del sitio Web un sistema gestión de contenidos educativos, que para el proyecto es MOODLE, ya que es software libre, compatible con MySQL, PHP y Apache.

Paralelo al diseño se procedio a desarrollar la solución Web y ha iniciar pruebas de los simuladores recopilados, para validar su funcionalidad y el acceso a los recursos que cada elemento del repositorio provea.

#### **4.3. Fase de identificación de la metodología educativa y el diseño de guías de implementación.**

Esta fase requirió que se definiera la metodología educativa a utilizar para validar el uso educativo de simuladores de física en un curso real. Se procedió a definir la población del curso, el nivel, y la modalidad educativa.

Seleccionado como grupo piloto el curso de Física I de la Tecnología en Obras Civiles de la Facultad de Estudios a Distancia de la UPTC.



29 de octubre al 9 de noviembre de 2008

Al mismo tiempo que se identifico esta metodología educativa de uso, se elaboro la guía de implementación del simulador como material educativo en un curso real, la cual debe ayudar a los alumnos a realizar de forma práctica una serie de experiencias de Física, utilizando el simulador seleccionado con el objeto de validar las fortalezas, debilidades y oportunidades de mejoramiento que permitan utilizar de forma más eficiente este tipo de materiales en la educación superior el aula.

De igual forma se definieron en esta fase los indicadores que permiten diagnosticar el uso educativo de los simuladores y el impacto de los mismos en la educación.

#### **4.4 Fase de aplicación**

Una vez que se ha identificado la metodología educativa, se han diseñado la guía de implementación y se tuvo disponibles los simuladores y el sistema de gestión de contenidos educativos se procedio a incorpora estos recursos a la práctica docente, como herramienta complementaria en la programación de trabajos prácticos de física, en los primeros curso de ingeniería en programas de pregrado de educación superior.

Y se procede a validar el uso de simuladores. También se busco analizar la influencia de estas aplicaciones informáticas, y de los materiales didácticos complementarios, en el proceso de aprendizaje que realizan los alumnos a través de las experiencias de laboratorio, evaluando los informes de las sesiones de trabajo realizadas con el software.

#### **4.5. Fase de diagnostico**

Esta fase final como su nombre lo indica busco establecer el diagnóstico frente a la validación del uso de los simuladores en la educación e identificar las fortalezas, debilidades y oportunidades de mejoramiento que permitan utilizar de forma más eficiente este tipo de materiales en el aula.

### **5. Enfoque Metodológico**

Es bueno precisar que el enfoque metodológico de esta investigación es describir la situación actual del tema, identificando los proyectos y desarrollos de simuladores, haciendo esta investigación de tipo descriptivo, teniendo como propósito primordial establecer el estado de desarrollo del tema uso educativo de simuladores desarrollados en software libre para física. Tener presente esta



29 de octubre al 9 de noviembre de 2008

consideración es fundamental para comprender los resultados que se presentan fruto de esta investigación.

Como uno de los objetivos es crear un banco de información que identifique y presente los desarrollos de software tipo simulador basados en software libre, dio lugar a establecer las características del análisis de información frente al tema y delimitando el marco general para la investigación. Las características que se han definido para la identificación de datos, su tratamiento y análisis son:

**Tipo de software:** característica centrada en los tipos de software libre o las condiciones de un software para denominarse software libre. Tomando como concepto que por lo menos cumpliera con una de las libertades que se han definido en el concepto de software libre.

**Licencia bajo la cual se distribuye el software,** se hizo necesario identificar esta característica para de esta manera presentar de forma más sólida los resultados.

**Lenguaje de programación,** esta característica busco establecer cuales lenguajes son los que lideran el desarrollo de simuladores en software libre.

**Plataforma o Sistema Operativo,** esta característica busco establecer en que sistemas operativos puede ser utilizado el software identificado

**Área del conocimiento,** aunque el proyecto se delimita para solo física, esta ciencia tiene múltiples áreas para su desarrollo, se identificaron las áreas fundamentales de la física y se busco asociar cada resultado a una de estas áreas.

Identificadas las características para hacer el levantamiento de información se hizo necesario categorizar cada una de ellas, definir variables y los indicadores que permitieran valorar los resultados.

Fruto de este trabajo para cada característica se tienen las siguientes variables

CARACTERISTICA	VARIABLES
Tipo de software	Web o Applet
	Aplicación instalable
	Sistema Operativo
	Directorio
	Entorno de desarrollo
	LMS Sistema gestor de contenidos o entorno virtual de aprendizaje
Lenguaje de programación	C++
	Java
	Python
	Perl
	Swish / Flash
Plataforma	GNU/Linux
	UNIX
	Windows





29 de octubre al 9 de noviembre de 2008

Tipo de licencia	Mac OS
	GNU GPL
	Freeware
	LGPL
	Creative Commons
	Shareware
	BSD

**Tabla 1.** Variables de Investigación

## 6. Análisis de resultados.

Para el proceso de documentación de los resultados el instrumento utilizado fue una hoja de registro, la cual se utilizó a lo largo del proceso de identificación y permitió analizar y procesar los datos obtenidos según las necesidades y objetivos definidos por el proyecto.

Nombre de la columna	Observaciones
Nombre del proyecto	Título del proyecto identificado
Dirección Web del proyecto	Registro el nombre de la página Web donde se aloja el proyecto.
País de origen	Lugar
Propósito	Descripción del proyecto
Año	Año de creación del proyecto
Lenguaje	Tipo de lenguaje utilizado en el desarrollo
Plataforma	Tipo o tipos de SO que soporta
Tipo Licencia	Tipo de licencia para el uso de los recursos
Categoría	Tipo de software identificado
Idioma	Lenguaje original y traducciones.
Áreas del Conocimiento	Se define las áreas y se identifica el número de desarrollos identificados.
Totales	No. de recursos identificados

**Tabla 2:** Ficha de registro de los estudios explorativos.

En síntesis proyectos de software educativo que sean netamente software libre son pocos, muchos de los resultados obtenidos luego de hacer la investigación



29 de octubre al 9 de noviembre de 2008

son soluciones de software que cumplen parcialmente el concepto de software libre, dado esto en términos de las cuatro libertades que el software libre debe cumplir.

Este trabajo ha permitido establecer tres grandes categorías de soluciones de software de simulación, a través de estas se hace la clasificación general de los resultados. Por otra parte y por el impacto que el proyecto pretende tener en cuanto a la física, se hizo necesario valorar e identificar el área dentro de la física que cada simulador aborda y se presentan los resultados en términos de número de simuladores identificados en cada área.

Finalmente la caracterización de los resultados toma como referente los tipos de software libre existentes y se analizan los resultados según estos tipos y la licencias con las que se distribuyen.

Es importante resaltar que se busco identificar el lenguaje en que se han desarrollado los simuladores, encontrando que el mayor porcentaje de trabajos están desarrollados en lenguaje Java, y porcentaje final se encuentran desarrollados en C o C++, Python e inclusive en Swish o Flash, este último distribuidos de forma libre pero sin acceso al código fuente.

El repositorio o banco de objetos de simulación puede consultare en el Web site denominado Open Simula. <http://www.ticagi.org>

### 6.1. Identificación de recursos por tipo de software

Este análisis permitió determinar la cantidad y el porcentaje de simuladores por clase de software identificado, con el objetivo de conocer la tendencia del mercado frente a los tipos de software.

Tipo Software	Cantidad Proyectos	%	No. Objetos	%
Applet (Java /Flash)	22	44,0	2069	98,7
S.O	1	2,0	1	0,1
Aplicación Ejec.	15	30,0	15	0,7
IDE	5	10,0	5	0,2
Directorio	2	4,0	2	0,1
LMS	5	10,0	5	0,2
<b>Total proyectos</b>	<b>50</b>		<b>2097</b>	

**Tabla 3:** Clasificación simuladores por tipo de Software

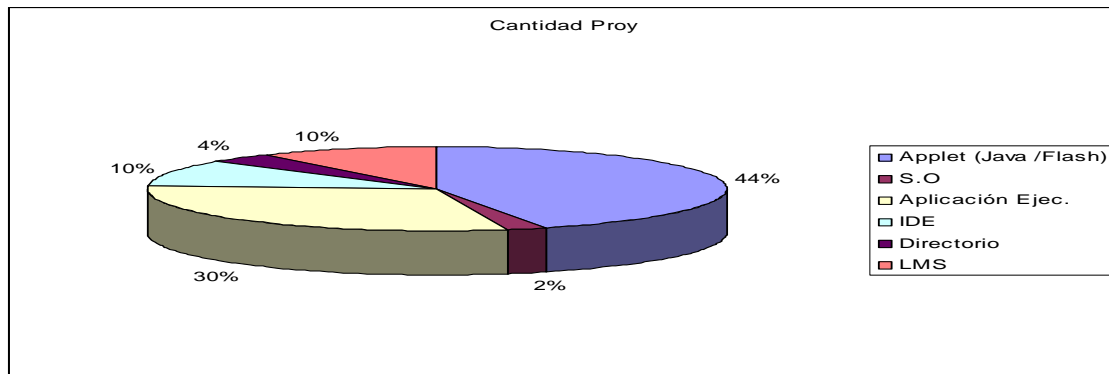
Analizando la tabla 3 se pudo establecer que los desarrollos tipo Applet son los más comunes en la simulación en software libre para física. La causa de este resultado esta en la portabilidad de los applets, su facilidad de ejecutarse sobre la



29 de octubre al 9 de noviembre de 2008

Internet, y la robustez para el manejo matemático de los lenguajes utilizados para estas simulaciones, Java y Flash.

La grafica 1. muestra como la población de simuladores de física desarrollados en software libre se distribuyen según el tipo de software.



**Grafica 1.** Porcentajes de simuladores según tipo de software

Los simuladores identificados tipo applet desarrollados en lenguaje Java, permiten descargar el código fuente o binario y los archivos de ejecución (.jar).

En materia de aplicaciones tipo ejecutable, se pudo establecer que corresponden en su gran mayoría a aplicaciones para simular gráficas de alta complejidad matemática.

Finalmente los entornos de desarrollo tipo IDE para creación de simulaciones en software libre son escasos pero poco a poco se consolidan como el futuro de los simuladores en software libre. En un numeral más adelante se describen con detalle estas soluciones.

A nivel de directorio Web o repositorios de recursos fue posible identificar dos repositorios de los más de 50 proyectos, estos repositorios son de tipo base de datos informativa.

Y a nivel de sistemas operativos de código abierto se identificó el sistema operativo Ing-X, que en su distribución integra los más robustos y complejos desarrollos para simulación. Este proyecto tiene sus orígenes en la república de la Argentina e integra más de 15 soluciones de simulación para múltiples áreas del conocimiento científico.

## 6.2. Identificación de recursos según el lenguaje de desarrollo

Dentro del proceso de investigación y levantamiento de los proyectos, a cada proyecto se le identificó el lenguaje de programación en el que fue desarrollado, el objeto de este análisis es establecer la tendencia en lenguajes de programación para el desarrollo de simuladores e identificar las causas de su uso.

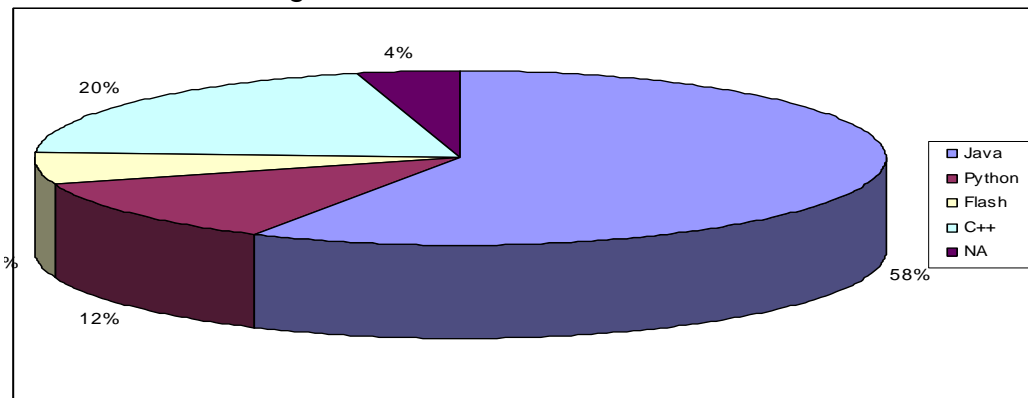


29 de octubre al 9 de noviembre de 2008

Lenguaje Programación	Cantidad Proyectos	%	No. Objetos	%
Java	29	58,0	1880	89,7
Python	6	12,0	7	0,3
Flash	3	6,0	198	9,4
C++	10	20,0	10	0,5
NA	2	4,0	2	0,1
<b>Total</b>	<b>50</b>		<b>2097</b>	

**Tabla 4:** Clasificación simuladores según lenguaje de programación

De forma concluyente este análisis arroja que el lenguaje Java es el más utilizado para el desarrollo de simuladores, seguido de Flash, C++ y Python. Este último lenguaje está siendo probado en el sistema SimPy, Simulation in Python, proyecto que ha creado una solución basada en este lenguaje para desarrollar simuladores en software libre, véase gráfica 2.



**Gráfica 2** Porcentajes de simuladores según lenguaje de programación utilizando

Los motivos por los cuales el lenguaje Java es el más utilizado en el desarrollo de simuladores para física son:

- Java es un lenguaje multiplataforma que permite desarrollar de forma independiente al sistema operativo las aplicaciones. Haciendo totalmente portable y escalable todo desarrollo en Java.
- Sus paquetes y APIs para manejo matemático y gráfico en 2D Y 3D permiten integrar y manejar las complejas operaciones y gráficas que la física requieren para simular los diferentes eventos y áreas de estudio.
- Java es un lenguaje modular altamente reutilizable, que permite crear un desarrollo y luego sin muchos cambios traducirlo, razón que ha hecho que en el mercado los simuladores tipo applet en java estén disponibles en múltiples idiomas.
- . Por otra parte su robustez en procesamiento matemático avanzado lo hace muy útil en el tema de simulación en física, al igual que los paquetes de clases para manejo de 2D y 3D y el manejo gráfico.
- Por otra parte al poder ejecutar un applet sobre un navegador de Internet, da lugar a que su distribución y uso se simple y funcione en prácticamente todo equipo de computo.



29 de octubre al 9 de noviembre de 2008

Algo muy similar sucede con Flash, es uno de los sistemas de desarrollo de mejor procesamiento gráfico y muy versátil a la hora de integrarse con la Internet, ya que estos desarrollos son orientados a la Web, a diferencia de Java solo es posible obtener en la gran mayoría de los casos el ejecutable mas no su código fuente, haciéndolo libre en uso pero no en las demás características definidas para el software libre

La diferencia central de los desarrollos en Java, C++ y Python frente a Flash esta en que los desarrollos bajo los 3 primeros lenguajes ofrecen las cuatro libertades establecidas para el software libre, mientras que Flash solo concede las libertades 1 y 3, ejecución y distribución.

### 6.3. Identificación de recursos según el tipo de Licencia de distribución

El objeto de incluir dentro de la caracterización del estudio este aspecto fue el de identificar la tendencia en materia de licenciamiento de los recursos tipo simulador identificados y el nivel de implementación de los compromisos del software libre y según la licencia de distribución de cada recurso o proyecto.

La tabla 5 refleja los resultados de este estudio, al analizarlos se puede ver con claridad que el Freeware como se le conoce al software que es gratis en cuanto uso y distribución más no en permisos para acceder al código fuente y cambiarlo o mejorarlo lidera por amplio margen los resultados.

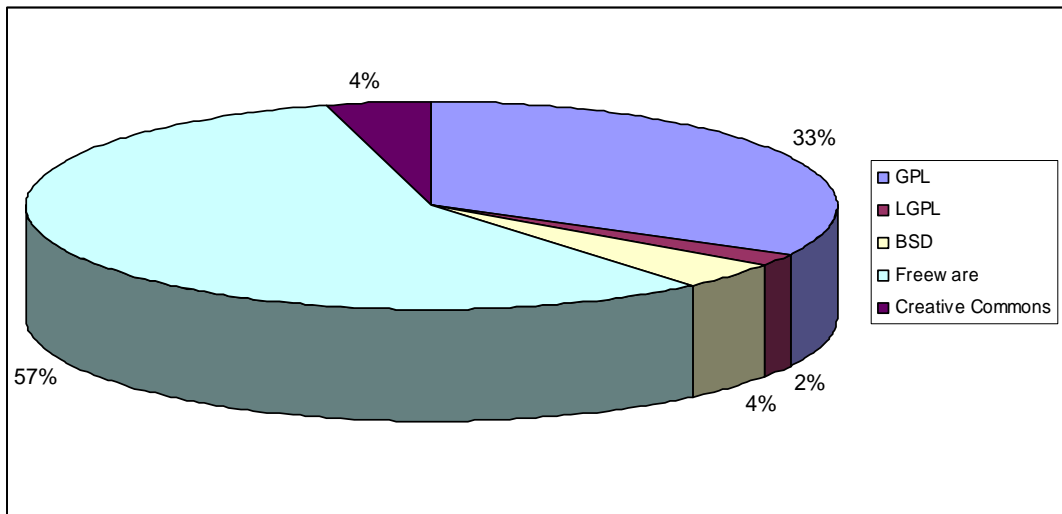
Se ve con preocupación que en este tema las libertades totales del software libre deben ser abordadas y concertadas a fin de aprovechar las fortalezas de las comunidades de desarrollo en el mejoramiento notable de estos recursos.

Tipo Licencia	Cantidad Proyectos	%	No. Objetos	%
GPL	16	32,7	120	5,7
LGPL	1	2,0	1	0,0
BSD	2	4,1	2	0,1
Freeware	28	57,1	1867	89,0
Creative Commons	2	4,1	107	5,1
<b>Total</b>	<b>50</b>		<b>2097</b>	

**Tabla 5:** Clasificación simuladores según las licencias de uso/distribución

El análisis lleva a concluir que el freeware en un 57.1 % es la forma mas común de publicar y poner a disposición del publico un software, y la desventaja del mismo es que no cumplen con las cuatro libertades establecidas por el software

libre, se ve de igual forma que la licencia GPL ocupa el segundo lugar con un 32.7 %, véase la grafica numero 3.



**Grafica 3:** Porcentajes de simuladores según tipo licencia con la que se distribuye

Esto lleva a que se plantee la necesidad de producir bajo un mismo criterio la forma de convertir el freeware en licencia GPL para garantizar las 4 libertades del software libre y potenciliar las ventajas que este modelo de desarrollo ofrece.

#### 6.4. Identificación de recursos según plataforma de uso

Necesariamente identificar la plataforma o sistema operativo sobre el cual puede ser usado el simulador, es fundamental. Esto asegura la transportabilidad entre uno y otro sistema operativo.

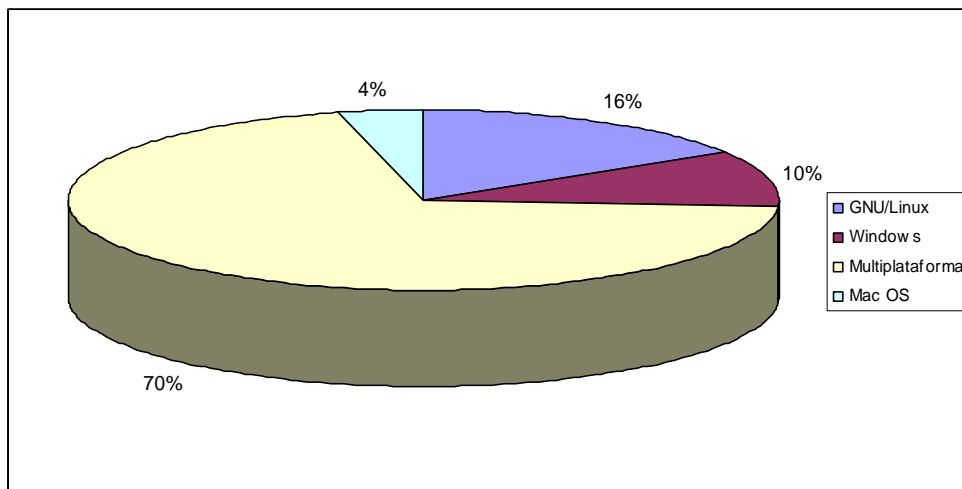
Plataforma	Cantidad Proyectos	%	No. Objetos	%
GNU/Linux	8	16,0	8	0,4
Windows	5	10,0	202	9,6
Multiplataforma	35	70,0	1885	89,9
Mac OS	2	4,0	2	0,1
<b>Total</b>	<b>50</b>		<b>2097</b>	

**Tabla 6:** Clasificación simuladores según los Sistemas operativos para su uso

Los resultados descritos en la tabla 6 de forma concluyente identifican que los desarrollos son en un 70% multiplataforma asegurando con ello el uso de los mismos tanto en GNU/Linux, como en Windows o incluso Mac OS. Ver grafica 4.



29 de octubre al 9 de noviembre de 2008



**Grafica 4:** Porcentajes de simuladores según su disponibilidad por Sistema operativo

Esto obedece a que la gran mayoría de soluciones están desarrolladas en java, y este lenguaje por ser portable permite que de forma muy sencilla pueda ser utilizado en múltiples

El caso de los sistemas operativos GNU/Linux sorprende por ser tan bajo, aunque al analizarlo con el hecho de pertenecer a la categoría multiplataforma ya lo está siendo en GNU/Linux de por sí.

## 7. Conclusiones y futuros trabajos

### 7.1. Conclusiones

En el trabajo se propuso identificar y documentar los proyectos de software educativo desarrollados en software libre mediante el desarrollo de un repositorio en línea, igualmente desarrollado en software libre.

Es de precisar que los resultados fruto de esta etapa del proyecto son satisfactorios en cantidad, se identificó alrededor de 2000 soluciones de software, cada uno documentado y organizado según el área de aplicación.

De este estudio se concluye que el software educativo que sean ciento por ciento libres son pocos apenas un 42.4% en número de proyectos identificados y en número de simuladores un 11.4% del total, situación que obedece a la no identificación de los autores con los modelos de licenciamiento de software libre, situación que lleva a concluir que en la actualidad el concepto de software libre bajo los principios que a este lo rigen aún no son claros para las comunidades de desarrollo y las instituciones.

Se requiere mayor esfuerzo por parte de las asociaciones y fundaciones gestoras y promotoras del software libre y los expertos en el tema para clarificar el concepto.



29 de octubre al 9 de noviembre de 2008

---

Por otra parte en su gran mayoría estos simuladores son iniciativas personales e individuales de profesores de diferentes instituciones que movidos por el deseo de mejorar la educación han desarrollado este tipo de recursos y que por la necesidad de los mismos han logrado vincular a estas iniciativas a otros colegas.

Se requiere liderar de forma más organizada y con criterios de calidad de software educativo y de la dinámica de sistemas este tipo de proyectos para que dentro de una comunidad de desarrolladores se consolide un modelo de desarrollo de software educativo libre.

Desde el punto de vista educativo se identificaron varios proyectos que en lo metodológico y pedagógico son muy sólidos que pueden ser los derroteros para consolidar el uso educativo de simuladores dentro de la sociedad.

## 7.2. Futuros trabajos

Identificadas más de 2000 soluciones de software desarrollados en software libre un trabajo futuro muy importante estaría dado por la validación en términos educativos de estos recursos y bajo los criterios de calidad del software educativo. Investigación que a las organizaciones educativas y docentes les permitiría tener mejores elementos a la hora de hacer uso de estos recursos.

De otra parte con la existencia de entornos integrados de desarrollo de software libre se debe realizar el análisis y estudio que lleve a definir la metodología para el desarrollo de simuladores y software libre utilizando estos sistemas y con sus respectivos procesos de validación desde lo pedagógico, el diseño instruccional y las teorías del aprendizaje que sustentan la calidad de estos sistemas.

Se plantea la posibilidad de investigar este mismo tema ampliándolo a otras áreas del conocimiento, ya que al desarrollar este proyecto fue posible vislumbrar que hay simuladores para muchas otras áreas, pero que de igual forma podrían identificarse y mediante un repositorio colocarlos al alcance de la comunidad.

Un proyecto posterior a este debe tomar estos simuladores y validarlos desde criterios que midan el aprendizaje y su impacto en los procesos de desarrollo de conocimiento.

## 8. Bibliografía

Castillo C. y Arrieta X. Referentes teóricos para el diseño y evaluación de software de apoyo a la enseñanza – aprendizaje de la física. Memorias de la IX Conferencia Interamericana de Educación en Física CIAEF-2006- IACPE. San José de Costa Rica. 3 a 7 de julio de 2006.

<http://www.efis.ucr.ac.cr/variopos/ponencias/9referentes%20teoricos.pdf>





29 de octubre al 9 de noviembre de 2008

---

Díaz-Antón, G. (2002) Uso de software educativo de calidad como herramientas de apoyo para el aprendizaje. Jornadas educativas: "La escuela como instrumento de cambio", IEA, Abril, Caracas. <http://www.academia-interactiva.com/articulos.html>

Esquembre F. Creación de simulaciones interactivas en Java: aplicación a la enseñanza de la Física. Pearson – Prentice Hall, España 2004.

Fernández E. Implantación de proyectos de formación en línea. Alfaomega-RaMa, España 2004.

Forrester J.W. System Dynamics and Learner-Centered-Learning in Kindergarten through 12<sup>th</sup> Grade Education. 1998.

Franco I, Álvarez F. Los Simuladores, estrategia formativa en ambientes virtuales de aprendizaje. Revista Virtual Universidad Católica ISSN 0124-5821. [http://www.ucn.edu.co/portal/uzine/volumen21/articulos/3\\_Investigaci%C3%B3n\\_simuladores.pdf](http://www.ucn.edu.co/portal/uzine/volumen21/articulos/3_Investigaci%C3%B3n_simuladores.pdf)

Gil, S. y Rodríguez, E. (2001). Experimentos de Física usando nuevas tecnologías. Buenos Aires: Prentice Hall / Pearson.

Giuliano M., Marchisio S. y otros. Diseño de simuladores para un curso a distancia dirigido a docentes de Física de distintas jurisdicciones de Argentina. Current Developments in Technology-Assisted Education (2006) <http://www.formatex.org/micte2006/pdf/2047-2051.pdf>

Kofman H. Integración de las funciones constructivas y comunicativas de las NTICs en la enseñanza de la Física Universitaria y la capacitación docente. Premio del II Concurso "Educación en la red". 2003. <http://www.educared.org.ar/concurso-2/resenia/pdf/04-kofman.pdf>

Marqués P. (1995). Software educativo: guía de uso y metodología de diseño. Editorial Estel. Barcelona.

Palacios J. Repilado F. Una alternativa metodológica para la realización de Los laboratorios virtuales de física general en las Carreras de ingeniería. Memorias Congreso TIC aplicadas a las Ciencias. 978-950-34-0369-3. [http://colos.fcu.um.es/TICEC05/TICEC05/56\\_564.pdf](http://colos.fcu.um.es/TICEC05/TICEC05/56_564.pdf)

Pontes, a. (2005). Aplicaciones de las tecnologías de la información y de la comunicación en la educación científica. Segunda parte: aspectos metodológicos. Revista eureka sobre enseñanza y divulgación de las ciencias, vol. 2, nº 3, 330-343.



29 de octubre al 9 de noviembre de 2008

Rosado L., Herreros J.R., Nuevas aportaciones didácticas de los laboratorios virtuales y remotos en la enseñanza de la física, junio 2005. [http://209.85.165.104/search?q=cache:JCAD80K0vJYJ:www.formatex.org/micte2005/286.pdf+%22Nuevas+aportaciones+did%C3%A1cticas+de+los+laboratorios+virtuales+y+remotos+en+la+ense%C3%B1anza+de+la+f%C3%ADsica,+%22&hl=es&ct=clnk&cd=1&gl=co&lr=lang\\_es](http://209.85.165.104/search?q=cache:JCAD80K0vJYJ:www.formatex.org/micte2005/286.pdf+%22Nuevas+aportaciones+did%C3%A1cticas+de+los+laboratorios+virtuales+y+remotos+en+la+ense%C3%B1anza+de+la+f%C3%ADsica,+%22&hl=es&ct=clnk&cd=1&gl=co&lr=lang_es)

Sierra, J.L. (2003). Estudio de la influencia de un entorno de simulación por computadores el aprendizaje por investigación de la Física en Bachillerato. Tesis Doctoral. Universidad de Granada.

Teodoro, V.T. (2003). Modellus: Learning Physics with Mathematical Modelling. Tesis postdoctoral. Universidade Nova de Lisboa. Abril de 2008. <http://modellus.fct.unl.pt/mod/resource/view.php?id=334>