

SLAM-C++ en la construcción de aplicaciones multimedia

SLAM-C++ to build multimedia applications

Antonio Membrides Espinosa, Angel Rilder Salazar Calderín, Abel Ernesto Lorente Rodríguez

Universidad de las Ciencias Informáticas

amembrides@uci.cu

Resumen

Los software educativos se caracterizan por el procesamiento de grandes volúmenes de información en varios formatos de presentación, así como la concepción de lineamientos pedagógicos, también en algunos casos se hace necesario el seguimiento a los estudiantes con trazas mediante las cuales se puede seguir el avance cognitivo de estos. Brindan además la posibilidad a los profesores de interactuar y personalizar el software adicionando documentos, creado sus propios ejercicios lo que contribuye a un mejoramiento y actualización del software. Teniendo en cuenta que todos estos elementos aumentan considerablemente la complejidad de este tipo de software, además de la necesidad de herramientas totalmente libres y robustas para la construcción de multimedia, se decide confeccionar un framework multiplataforma para el desarrollo de multimedia educativa, el cual fue denominado SLAM-C++.

En el presente trabajo se exponen los resultados de todo el proceso investigativo, así como las razones por las cuales es de vital importancia desarrollar herramientas propias para la construcción de productos multimedia. Con el fin de lograr este objetivo se desarrolló un análisis comparativo acerca de las tendencias y tecnologías libres existentes utilizadas en la construcción de estas aplicaciones. Posteriormente se procede a conceptualizar el sistema propuesto.

Palabras Claves: Framework, Multimedia, Sistemas de Autor, Software Libre

Abstract

The educational software is characterized by the processing of large volumes of information in several formats. These systems are also defined by the design of educational tracks or the development traces of use, progress and cognitive user involvement to use databases and the ability of teacher to add multimedia elements to the system to enrich the software. All these factors increase the complexity of this type of software, besides the need for free and robust tools for building multimedia, it was decided to establish a framework for cross-platform development, which was called SLAM-C++.

In the present work is outcome of the entire investigative process, as well as the reasons why it is vital to develop its own tools for building multimedia products. We developed a comparative analysis on trends and technologies available free used in the construction of such applications. Subsequently proceeded to conceptualize the proposed system and the modeling of reach.

Key words: *Author's Systems, Framework, Free Software, Multimedia*

Introducción

La utilización de software propietario para el desarrollo de aplicaciones multimedia establecido en la industria del software nacional, garantiza gastos insostenibles y una creciente dependencia tecnológica hacia las transnacionales, siendo esta una de las razones por la que países como Cuba donde el desarrollo de las industrias tecnológicas es aún incipiente, presentando serias dificultades, no sería económicamente factible el uso de los mismos.

Frente a esta realidad, esta clara la necesidad de alternativas para la construcción de multimedia con tecnologías libres y sin

embargo es también la principal problemática. En la actualidad no se cuenta con sistemas de autor que se ejecuten sobre GNU/Linux para el desarrollo de aplicaciones multimedia y las herramientas que existen no cumplen en su totalidad con los requerimientos que implican estos productos.

Muchos han sido los esfuerzos por desarrollar herramientas libres análogas a Adobe Flash, sus mayores logros están dirigidos al entorno gráfico, sin embargo estas carecen de la capacidad para crear vectorizaciones complejas o interpolación de movimientos, con interfaces poco intuitivas, no adoptan en su plenitud una metáfora de trabajo específica para el desarrollo de este tipo de aplicaciones y ni siquiera soportan código script. Por otra parte se han desarrollado un conjunto de aplicaciones dirigidas al desarrollo desde el punto de vista del programador, como son compiladores y algunas herramientas que resuelven problemas específicos como elementos en las biblioteca, inyección de código, conversión de formatos, etc.

El problema no se fundamenta en la inexistencia de herramientas, sino todo lo contrario, en la actualidad la comunidad de software libre cuenta con una suma considerable de software que de una forma u otra permiten la creación de aplicaciones multimedia, sin embargo ninguno por separado puede ser considerado como una alternativa de desarrollo y su integración podría llegar a ser extremadamente compleja.

Partiendo de esta problemática en el año 2008 dos estudiantes de la Universidad de las Ciencias Informáticas bajo la tutoría de uno de los profesionales que pertenecía a la dirección de software educativo del centro, se plantearon la necesidad de elaborar una solución lo suficientemente sólida y robusta que permitiera desarrollar aplicaciones multimedia con tecnologías no propietarias, de forma tal que las mismas pudiesen ejecutarse sobre plataformas de software libre como GNU/Linux.

Desarrollo

Metodología Computacional

A lo largo de esta investigación se utilizarán un conjunto de métodos científicos que servirán de guía y facilitarán un mejor entendimiento de lo que está sucediendo, como es el caso de los métodos empíricos y los teóricos. A continuación se explica en detalles el por que de su selección.

Métodos Empíricos:

1. Encuestas y Entrevistas: Se entrevistarán a varios líderes de proyecto de software educativo en la UCI, diseñadores experimentados, así como a directivos de producción en la Universidad y del MINED, para recopilar toda la información que en estas puedan suministrar y de esta forma conocer cómo se desarrolla el proceso de desarrollo de software educativo de forma general en la Universidad y el país.

Métodos Teóricos:

1. Histórico-Lógico: Se llevará a cabo un análisis de la trayectoria hasta la actualidad del proceso de desarrollo de software educativo en nuestro país y fundamentalmente en la UCI, teniendo en cuenta las herramientas para el desarrollo de aplicaciones multimedia que se han utilizado, para conocer con mayor profundidad, los antecedentes y las tendencias actuales. Esto dará un preámbulo de que es lo que está pasando y un punto de partida para proponer posibles soluciones.
2. Modelación: Se desarrollará un conjunto de diagramas que propiciarán un mejor entendimiento de la arquitectura y de lo que se desea implementar.
3. Analítico-sintético: Se llevará a cabo un análisis de las teorías, documentos, etc, permitiendo la extracción de los elementos más importantes que se relacionan con el objeto de estudio.

Soluciones existentes para el desarrollo de multimedia

En la actualidad el desarrollo de herramientas libres para la construcción de aplicaciones multimedia no es lo suficientemente sólido como para saciar las necesidades que estos productos implican, la comunidad de software libre ha empleado todo su empeño en desarrollar dichas herramientas, aunque todavía no se cuenta con sistemas a la altura de Macromedia, existen varios proyectos muy prometedores y ya se han empezado a ver los resultados. En el desarrollo de herramientas para la construcción de multimedia que corran sobre plataformas libres podemos identificar 2 vertientes:

1. Línea de desarrollo basada en formato SWF.
2. Línea de desarrollo basada en lenguajes de propósito general.

Dentro de cada una de estas líneas se identifican al menos 2 alternativas para desarrollar aplicaciones multimedia con tecnologías libres, véase la figura 1.

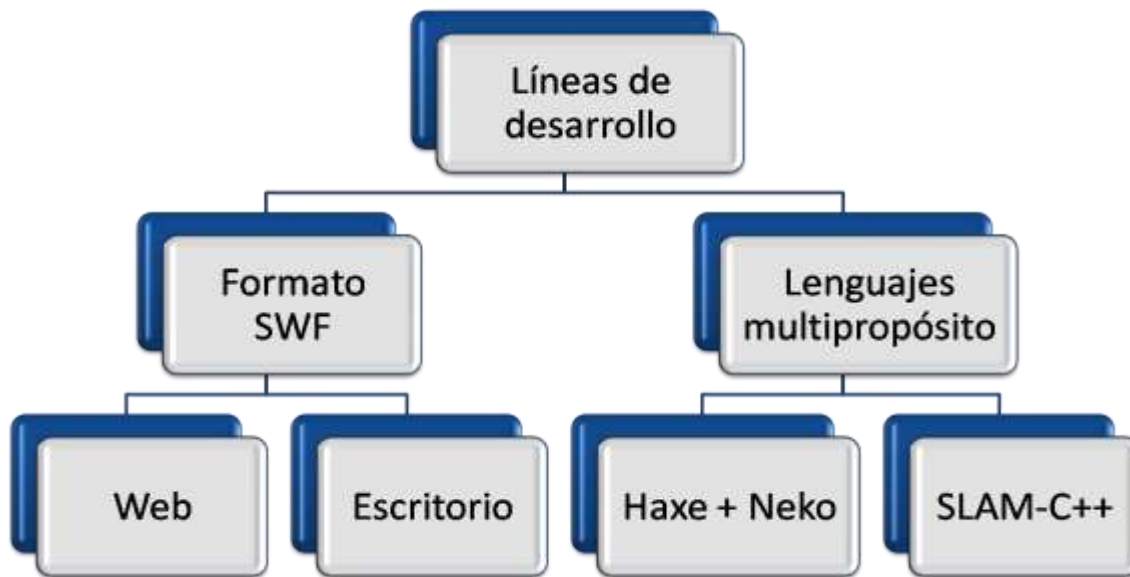


Fig. 1. Alternativas para el desarrollo de multimedia con tecnologías libres.

¿Por qué utilizar SLAM-C++?

Como bien queda representado en la figura anterior, una de las alternativas consiste en hacer uso de un conjunto de herramientas que faciliten el trabajo con el formato SWF, como por ejemplo SWFMILL, SWFTools, SWFExtract, etc, conjuntamente con el Adobe Flash Player 9 en su versión para GNU/Linux o el GNash para aplicaciones de escritorio. Contrario a esto podríamos reproducir los binarios SWF directamente para la Web, siendo la opción menos recomendada debido a la carga de información de todo tipo característica de aplicaciones multimedia, pues la eficiencia y rapidez de los productos finales no sería la óptima, incluso requeriría mucho más hardware de las PC de los futuros clientes tentado en su contra.

De una forma u otra las dos alternativas se basan en el formato SWF, si bien es cierto que el ActionScript provee de muchos mecanismos para el tratamiento de la animación con gráficos vectoriales, no soporta funcionalidades para acceso a disco duro, bases de datos, escritura en ficheros y XML, ni mecanismos para la programación multihilo, siendo esto elementos importantísimos para la construcción de productos multimedia de forma eficiente y con la calidad requerida. También es importante tener en cuenta que a pesar de la reciente liberación de formato SWF, no ha dejado de ser privativo y al utilizar el software de Adobe Systems Incorporated o cualquiera de sus filiales, se deben aceptar los términos y condiciones de usos que establecen sus licencias, o de lo contrario no se podría trabajar con sus productos.

Un elemento importante en el uso de Neko y Haxe para desarrollar multimedia, es su extensibilidad, permitiendo cargar librerías compartidas hechas en C, de forma tal que aquello que no se pueda hacer con el lenguaje Neko, se podría solucionar con C. Claro esta que si se tienen que desarrollar mediante librerías compartidas la mayoría de funcionalidades necesarias para cumplir con los requerimientos que implican las aplicaciones multimedia, entonces no tendría mucho objetivo usar estos proyectos que son muy jóvenes y están sujetos a cambios, pudiendo utilizar un lenguaje multipropósito más completo y eficiente. El inconveniente que esto puede traer consigo estaría dado por el tiempo de desarrollarlo de SLAM-C++, sin embargo a largo plazo resultara más beneficioso teniendo en cuenta que se desarrollará una herramienta totalmente libre y multiplataforma que permitirá alcanzar verdadera independencia tecnológica en el campo de las aplicaciones multimedia.

¿Qué es SLAM-C++?

El propósito de SLAM-C++ es convertirse en un framework para el desarrollo aplicaciones multimedia a través de la Metáfora Cine. Al igual que se dirige una película, se podrán integrar elementos gráficos, conjuntamente con sus transformaciones realizadas sobre el escenario para producir transiciones y efectos especiales. Para ello se implementarán un conjunto de interfaces muy similares al modelo conceptual de dicha metáfora.

Las animaciones se basarán principalmente en la utilización de gráficos vectoriales, definiéndose el SVG como el formato soportado para este fin, también será posible trabajar con imágenes en diferentes formatos. Aunque se pueden crear los SVG a nivel de programación no es objetivo del sistema, se aconseja que se utilicen editores gráficos para su desarrollo, evitando de esta forma un aumento excesivo en el tamaño del ejecutable y el rendimiento del producto. Esto posibilita cierta independencia de roles en el desarrollo de aplicaciones multimedia, donde los programadores se centran en la lógica del negocio y los diseñadores en la interfaz gráfica. SLAM-C++ proveerá mecanismos sencillos para crear efectos y animaciones complejas, de forma tal que se puedan asociar interpolaciones de movimiento, forma y transparencia a determinado actor en el escenario para que este asuma dicho comportamiento. Para el trabajo con archivos de audio y video se utilizará un conjunto de interfaces que simplifica su utilización, apoderándose de las funcionalidades que brinda el reproductor mplayer, así como todos los formatos que este soporta. Por otra parte para la gestión de los comportamientos se utilizará el mecanismo de señales y ranuras que implementa QT. El sistema proveerá mecanismos para el tratamiento de datos. Esto incluye trabajo con XML, ficheros binarios y de texto, así como acceso a bases de datos soportando MySQL y PostgreSQL. Otro elemento importante a tener en cuenta es el paquete de clases para acceder y crear librerías dinámicas compartidas, posibilitando la confección de productos multimedia con arquitecturas basadas en componentes.

El framework deberá constar de tres herramientas principales:

- ✓ SLAM_Designer,
- ✓ SLAM_DCompiler
- ✓ SLAM_PCompiler.

SLAM_Designer ofrece una interfaz visual para el diseño gráfico, proveyendo los elementos conceptuales de la Metáfora Cine para la construcción de multimedia, tal es el caso de la línea de tiempo, componentes para trabajo con texto, gráfico vectorial, imágenes, bases de datos, etc. Facilitando de esta forma el desarrollo de aplicaciones multimedia con SLAM-C++. Para ello esta herramienta generará un archivo en formato XML con la configuración del entorno gráfico multimedia.

SLAM_DCompiler es la herramienta encargada de leer los ficheros XML generados por SLAM_Designer y crear las clases correspondientes incluyendo sus dependencias, en el lenguaje C++ utilizando las librerías de SLAM-C++.

SLAM_PCompiler se encarga del proceso de compilación del framework ejecutando las herramientas e instrucciones necesarias,

abstrayendo al programador de este tedioso proceso y facilitando la integración de SLAM-C++ con algún IDE.

Formatos soportados en el framework:

- ✓ **Formatos de Gráficos Vectorial:** SVG
- ✓ **Formatos de Mapas de Bit:** JPEG/PNG/GIF/PGM/BMP/DIB
- ✓ **Formatos de Audio:** MP3/WAV/WMA/MP2
- ✓ **Formatos de Video:** MPEG/VOB/IFO/AVI/WMV/ASF/OGG/OGM/MKV/MP4/DAT/RM/RMVB

El framework costa de 6 módulos principales que engloban todas las funcionalidades y servicios que brinda el sistema, véase figura 2.

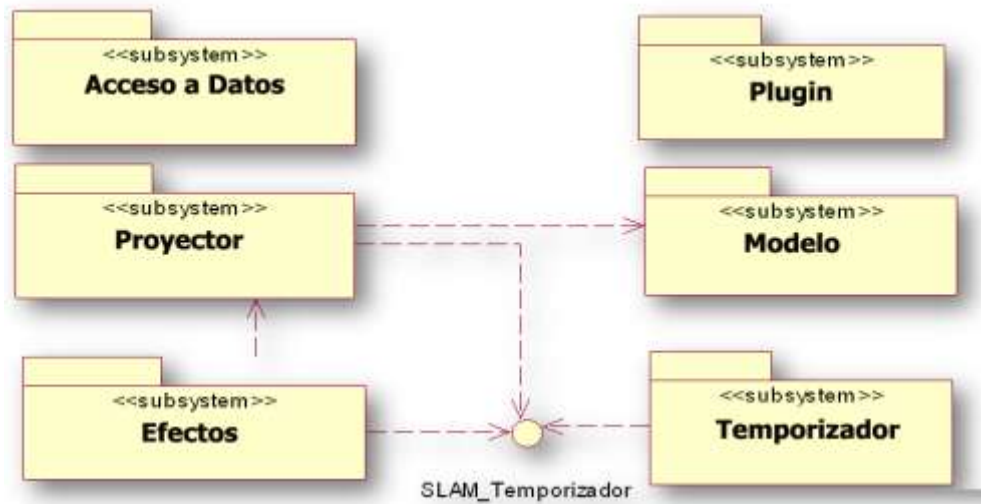


Fig. 2. Módulos del framework de SLAM-C++.

- ✓ El módulo **Acceso a Datos** facilita el trabajo con Ficheros, XML y Bases de Datos, siendo este un elemento importante en la creación de productos multimedia y al mismo tiempo una de las principales deficiencias de las herramientas libres.
- ✓ El módulo **Proyector** provee de un conjunto de clases que implementan la Metáfora Cine, abstrayendo de esta forma a los desarrolladores del proceso de construcción de multimedia.
- ✓ El módulo **Plugin** provee mecanismos para desarrollar aplicaciones multimedia aplicando arquitecturas de componente, un elemento importantísimo que facilita la reutilización de código y el soporte de las mismas.
- ✓ El módulo **Efectos** provee mecanismos para desarrollar animaciones complejas muy similares a las que se pueden utilizar en Adobe Flash, tales como interpolación de movimiento, de forma y transparencia.
- ✓ Los módulos **Temporizador** y **Modelo** son totalmente transparentes al programador, pues estos son utilizados directamente por el modulo Proyector, estos se encargan respectivamente de las transición o secuencia de los objetos en el tiempo y de los elementos que serán cargados en la aplicación multimedia.

Tecnologías

En el proceso de construcción del framework SLAM-C++ se emplearan un conjunto de tecnologías que tributen a los objetivos propuestos, es por ello que el framework esta dirigido al lenguaje de programación C++, teniendo en cuenta su integración con C y el soporte de instrucciones en Ensamblador, permitiendo aprovechar al máximo los recursos de hardware, así como programación orientada a objetos y la modularidad del código que el mismo provee.

Es importante tener en cuenta que trabajar con una tarjeta gráfica es extremadamente complicado a nivel de programador, en consecuencia a esto han surgido interfaces que abstraen la complejidad y diversidad de las primitivas del hardware, es por ello que

en la implementación de las primeras versiones de SLAM-C++ se utilizará el framework QT, aprovechando la arquitectura para plugins que este provee, así como mecanismos para la programación multihilo y comunicación entre procesos, funciones para Red, XML, multimedia, manejo de archivos, librerías dinámicas, librerías básicas Entrada/Salida, interface con bases de datos como Oracle, MySQL, PostgreSQL, ODBC, además de la excelente documentación y la disponibilidad del código fuente. Otro elemento importante a tener en cuenta es el soporte de aplicaciones embebidas, permitiendo integrarse fácilmente con otros software para extender sus funcionalidades, tal es el caso de la herramienta mplayer el cual será empleado para el tratamiento de audio y video.

Por otro lado se utilizará RUP como metodología de desarrollo de software que registrará el proceso de construcción del mismo y el UML como lenguaje de modelado, teniendo en cuenta que SLAM-C++ constituye un proyecto de gran envergadura y se requiere de una metodología capaz de documentar todo el ciclo de desarrollo. Para un mayor entendimiento de las tecnologías empleadas véase la figura 3.

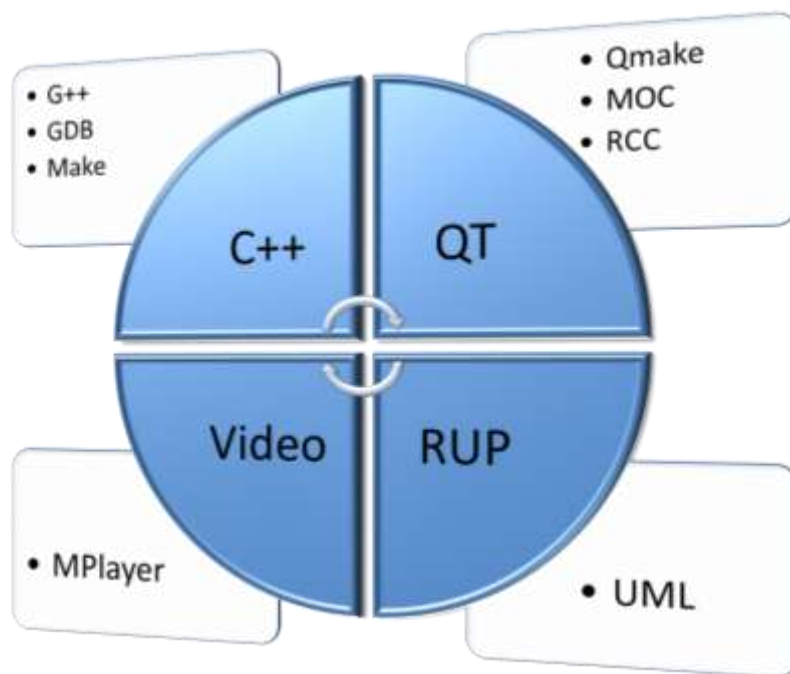


Fig. 3. Herramientas, tecnologías y metodologías en el desarrollo de SLAM-C++.

Arquitectura

La arquitectura de software es la organización fundamental de un sistema encarnado en sus componentes, las relaciones entre ellos y el ambiente, así como los principios que orientan su diseño y evolución, es por ello que en el desarrollo de SLAM-C++ no se paso por alto el estudio de la misma y se definió la arquitectura Orientada a Objeto como la que registrará el sistema.

Como bien se había expresado en tópicos anteriores SLAM-C++, provee un conjunto de capas de abstracción para facilitar el trabajo de los desarrolladores, e independizarse de las plataformas de hardware y software, de forma tal que sus productos sean multiplataforma. Para un mayor entendimiento de cómo esta conformado el framework SLAM-C++ en términos de capas de abstracción, véase la figura 4.

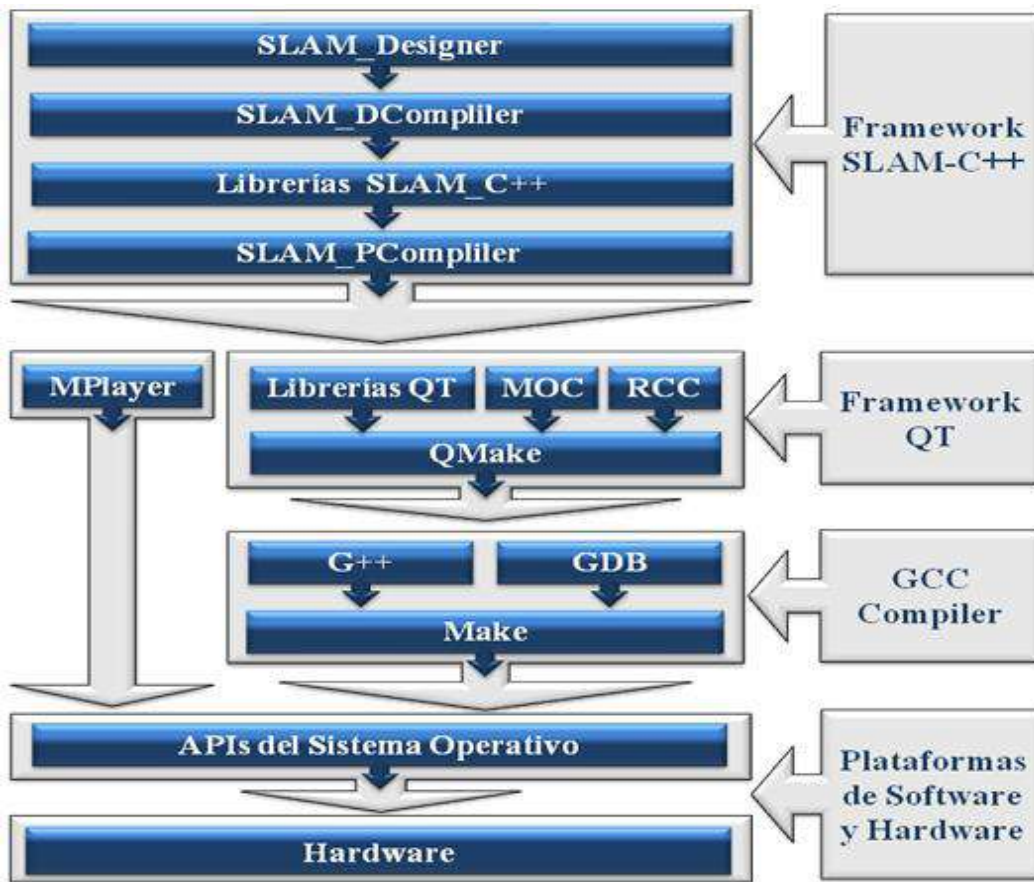


Fig. 4. Vista externa del framework SLAM-C++ en términos de capas de abstracción.

Metamorfosis

El desarrollo de aplicaciones multimedia es extremadamente complejo y aún utilizando un framework para su construcción, estarían presentes muchas limitantes en cuanto al tiempo de desarrollo y el coste de aprendizaje del mismo. Un elemento decisivo en la viabilidad de un proyecto es el tiempo de desarrollo, los sistemas de autor han jugado un importantísimo papel en este campo, proveyendo un conjunto de mecanismos e interfaces enfocados en un único objetivo que agilizan dicho proceso.

SLAM-C++ constituye el punto de partida para lograr un genuino sistema de autor. Este está concebido para que en un futuro pueda integrarse con algún IDE existente. En la actualidad existen un gran número de entornos integrados de desarrollo que compilan código C++ estándar, facilitándose de esta forma el proceso de compilación del sistema propuesto, el problema radicaría en la integración de la herramienta SLAM_Designer con el supuesto IDE, sin embargo tanto KDE como QT proveen de un mecanismo conocido como aplicaciones embebidas que resuelven dicha problemática, aunque lo ideal sería fusionarlo con un IDE nacional, logrando de esta forma la verdadera independencia tecnológica. Véase la figura 5.



Fig. 5. Metamorfosis de SLAM-C++ en un Sistema de Autor.

Conclusiones

Este proyecto es totalmente innovador, que en la actualidad no se ha desarrollado nada semejante, lo que se propone consiste en aprovechar los recursos que brindan los lenguajes de alto nivel, conjuntamente con las librerías de interfaz de usuario para recrear una metáfora de trabajo que establezca un conjunto de capas de abstracción de forma tal que le permita a los programadores desarrollar aplicaciones multimedia de forma sencilla y sin necesidad de tener elevados conocimientos relacionados con la plataforma para la que se desea desarrollar dicho sistema.

Al finalizar el presente trabajo, se consideran cumplido el objetivo trazado y suficientemente fundamentado. A este planteamiento se puede arribar a partir de las siguientes conclusiones:

- ✓ Una de las principales metas del estado cubano es la independencia tecnológica, por tanto el proceso de migración a software libre no solo constituye una necesidad, sino que se convierte en una estrategia que permitirá sentar las bases de forma tal que pueda crear sus propios software y por consiguiente es prioridad el desarrollo de herramientas para la construcción de aplicaciones multimedia que corran sobre plataformas libres como GNU/Linux, lo que permitirá agilizar dicho proceso en la esfera de la educación.
- ✓ Debido a la inexistencia de herramientas sólidas para la construcción de aplicaciones multimedia sobre la plataforma GNU/Linux y a la complejidad de integración de las existentes, resulta más factible la construcción de un framework multiplataforma que resuelva dicho problema.
- ✓ El framework esta orientado a la construcción de multimedia a través de gráficos vectoriales y no de imágenes raster, evitando un aumento excesivo en los binarios ejecutables.
- ✓ El framework se basa en el cargado del contenido multimedia y no en la creación de estos recursos.
- ✓ La utilización de este framework rompe en cierta medida con los enfoques tradicionales para el desarrollo de productos multimedia. En la actualidad es difícil lograr productos con calidad y que sean altamente reutilizables, teniendo en cuenta que el éxito en la creación de aplicaciones depende del nivel organizacional y la solidez de su estructura interna, SLAM-C++ facilita el desarrollo de este tipo de aplicaciones, pues este garantiza el diseño orientado a objetos facilitando de esta forma la reutilización de código.
- ✓ SLAM-C++ constituye las bases de un futuro Sistema de Autor.

Referencias Bibliográficas

Adobe Systems, Adobe, 2008, <http://www.adobe.com/>.

Antonio Membrides Espinosa y Angel Rilder Salazar Calderín. SLAM-C++. UCI. Ciudad Habana, 2008. Pág. 26, 47, 73, 74, 75, 94, 102, 103 y 105.

Ministerio de Relaciones Exteriores de la República de Cuba, Cuba MinRex, 2008
<http://www.cubaminrex.cu>.

Motion Twin, Haxe, 2008, <http://haxe.org/es>.

Motion Twin, Neko, 2005, <http://nekovm.org/>.

Nokia Corporation , Qt Reference Documentation, 2008, <http://doc.trolltech.com/4.4/index.html>

Trolltech, QT, 2008, <http://doc.trolltech.com>.