

I Concurso Universitario de Software Libre PROYECTOS

El Concurso Universitario de software libre, organizado por la Escuela Técnica Superior de Informática de la Universidad de Sevilla, entra en su recta final. El avanzado desarrollo de los proyectos presentados nos indica que estamos próximos a conocer los ganadores de cada una de las categorías.

Los organizadores del concurso y Linux Magazine hemos escogido unos cuantos proyectos para que sus artífices expliquen con sus propias palabras en qué consisten, cómo va el desarrollo y los pormenores de su aplicación.

En esta entrega vemos un traductor al estilo "Babylon", un juego para DS para aprender Kanjis japoneses y un proyecto para una "flauta" electrónica con su correspondiente MIDI para Linux.

CoolTran

Por Javier García Palacios y Diego Sevillano de Miguel

Antes de nada, queríamos agradecer a "Linux Magazine" su interés por nuestro proyecto y por todos los proyectos de software libre, pues creemos en el software libre y queremos ser partícipes de esa filosofía, en un mundo donde las patentes, cánones y derechos de copyright se están volviendo cada vez más agresivos y cuestionables.

Queremos hacer referencia a un artículo muy interesante de Richard Stallman que hace alusión a esto mismo: "¿Puede confiar en su computadora?"

Búsqueda de proyecto

Cuando empezamos a buscar un proyecto, ignorábamos el alcance de posibilidades que éste ofrecía. Pero en nuestra búsqueda topamos con el despacho de nuestro tutor actual, Francisco Javier Pérez, que nos informó de la existencia de un concurso universitario de software libre a nivel nacional, de forma que nuestro proyecto no sólo pudiera ser de gran utilidad, sino que con él también pudiéramos participar directamente en esa comunidad de software libre. Así que nos invitó a pensar en alguna aplicación o programa que nos gustase hacer.

La idea

Tras mucho pensar, y como usuarios de Linux, buscamos aplicaciones o herramientas para Linux que no estuviesen hechas ya, o que se pudiesen mejorar e, investigando, encontramos que no había un programa equivalente al Babylon para Windows y que utilizase además la tecnología OCR (*Optical Character Recognition*, Reconocimiento Óptico de Caracteres), para no tener que seleccionar la palabra entera.

Así pues, nuestra propuesta consiste en realizar un programa de software libre para Linux, de funcionalidad similar al Babylon, que al "clicar" una palabra en otro idioma con el ratón, salga una ventana pop-up o similar con su traducción, utilizando para ello un OCR abierto y diccionarios de idiomas en local o vía Internet.

Además, pensamos que, ya que la aplicación necesitaba al menos un diccionario de idiomas para traducir las palabras, ¿por qué no hacer un diccionario colaborativo en el que los usuarios pudieran valorar las traducciones e incluso añadir otras nuevas mediante la misma aplicación o mediante una página web?

Desarrollo

Para el desarrollo de nuestro proyecto decidimos utilizar el lenguaje de programación Java, ya que, tanto por sus características (por-

table, multiplataforma, orientado a objetos) como por la reciente apertura de su código creemos, que es la mejor opción de cara al futuro.

Dado que actualmente no todo el código de Java ha sido abierto, para el desarrollo de nuestro proyecto es posible que necesitemos utilizar alguna de sus implementaciones libres, como las librerías GNU Classpath.

Realizar también el diccionario colaborativo puede que exceda los objetivos del proyecto fin de carrera, pero si nos da tiempo, éste consistirá en una base de datos (probablemente en MySQL) en Internet a la que usuario pueda acceder a través de la aplicación y/o mediante una página web.

CoolTran

La aplicación funcionará de la siguiente manera: el usuario está trabajando en su ordenador o bien navegando por Internet, leyendo un documento de texto o lo que sea, se encuentra una palabra en otro idioma cuyo significado desconoce. Entonces, mediante una combinación de teclas y ratón (como por ejemplo: ctrl + clic izqdo) sobre la palabra deseada, se lanza la aplicación que obtiene el término de la captura con el OCR. A continuación se busca su traducción, bien en el diccionario local, si está instalado, o bien en el diccionario colaborativo en Internet. La palabra reconocida por el OCR y el resultado de la traducción se muestran en una pequeña ventana, por ejemplo, en la esquina inferior derecha de la pantalla, donde también se puede mostrar la búsqueda del término en otros diccionarios de Internet como Freedict, Google, Wikipedia...

Una de las cosas más atractivas de nuestra aplicación, bajo nuestro punto de vista, es su sencillez y claridad, sustituyendo la clásica barra de menús por pestañas. La primera pestaña, la principal, es la búsqueda del término, la segunda es la de configuración y selección de los diccionarios de idiomas, la tercera es la de configuración de la aplicación, idioma de la aplicación, combinación de teclas para buscar un término, etc. Existirá una última pestaña de ayuda e información acerca del programa.

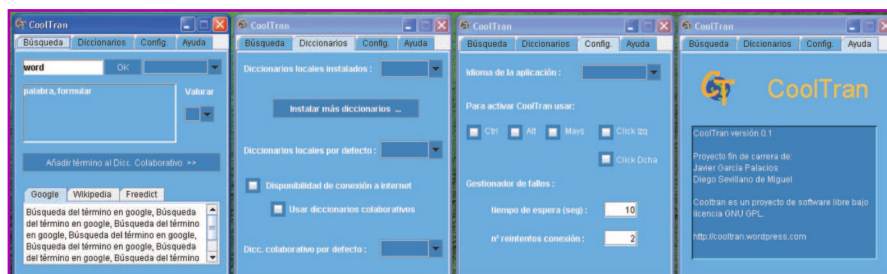


Figura 1: Esquema conceptual de OpenPipe.

Más información

Actualmente puede encontrarse información del desarrollo de nuestro proyecto en la Forja-RedIRIS en [1], así como en nuestro blog en [2], donde exponemos hitos del desarrollo, capturas de pantalla de la interfaz gráfica, etc.

Kana DS

Por Julio Gorgé

Aprende Japonés con tu Nintendo DS.

Descripción del proyecto y motivación

Kana DS es una aplicación de ayuda al aprendizaje del idioma Japonés para Nintendo DS con la que participo en el Iº Concurso de Software Libre.

Como casi cualquier otro joven que estudia japonés, mi interés por la cultura nipona proviene principalmente del manga, el anime y los videojuegos. Supongo que el Japonés debe estar de moda. La gran afluencia al Salón del Manga en Barcelona cada año, el éxito del blog de Kirai, y el hecho de que ya se impartan clases de Japonés en 17 universidades españolas son ejemplos de la popularidad de la cultura nipona en España.

La cuestión es que cuando comencé a estudiar el idioma hace unos meses, encontré mucho software para aprender japonés en PDA o PC. Sin embargo, casi todos los programas se centraban en la ayuda a la memorización sistemática de los 2000 ideogramas de uso común, con lo cual dejaban fuera a los principiantes. En pocas palabras, el software era aburrido. Los japoneses han reconocido el potencial de Nintendo DS para herramienta de aprendizaje y disponen de diccionarios y juegos que enseñan kanji a los más pequeños.

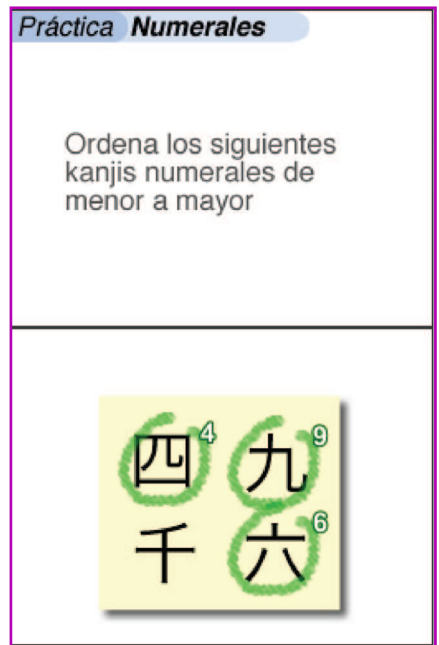


Figura 2: Las pestañas de Cooltran.

Sin embargo, dichos títulos están íntegramente en japonés y no han sido desarrollados con el mercado occidental en mente.

¿Por qué Nintendo DS?

Porque es posible crear juegos para aprender idiomas sin que resulten aburridos. English Training para Nintendo DS es un ejemplo, y mi intención es hacer algo similar con Kana DS.

¿Pero Nintendo DS no es una plataforma cerrada? En efecto. Por eso en un principio consideré la posibilidad de desarrollar para el Nokia 770, ya que usa GNU/Linux como sistema operativo y cuenta con una pequeña comunidad de código libre a su alrededor. Sin embargo, quería que Kana DS pudiera ser disfrutado por el mayor número de gente, y ni el precio del Internet Tablet ni su reducida base de usuarios contribuyen a ello.

En cambio, a día de hoy ya se han vendido más de 10 millones de DS en toda Europa, y todo el mundo la conoce y la sabe usar.

Entonces, ¿qué pasa con la naturaleza cerrada de la consola? Aunque los kits de desarrollo oficiales están reservados a las empresas, es posible realizar aplicaciones y juegos mediante kits no oficiales desarrollados por aficionados mediante ingeniería inversa. Sólo hace falta uno de los múltiples flash-cards (cartuchos flash) que se venden a través de Internet e introducir nuestro programa en él. Así que lo primero es dar las gracias a estos hackers por su trabajo, que hacen posible que podamos desarrollar nuestro propio software para Nintendo DS. Concretamente yo uso la devkitPro y libnds.

Contenidos del juego

En Kana DS suponemos que el jugador ha comenzado a estudiar japonés recientemente, por lo que aún no domina a la perfección el silabario hiragana ni el katakana, por no hablar del centenar de ideogramas -o kanji- más básicos. Los silabarios corresponderían a nuestro alfabeto, y los kanji se usan para representar palabras o conceptos comunes. Los niños japoneses comienzan aprendiendo la escritura del hiragana y, más adelante se introducen el katakana y los kanji.

El objetivo del juego es precisamente reforzar la lectura y escritura de los dos silabarios y los 80 kanji básicos mediante la repetición de pequeños juegos. Para que realmente se experimente un progreso adecuado, el nivel de los juegos ofrecidos en cada momento estará acorde al nivel del jugador e irá aumentando en dificultad según su ritmo de aprendizaje.

Uno de los ejercicios del juego es la práctica de numerales: En pantalla aparecen varios kanji asociados a números, y el jugador debe

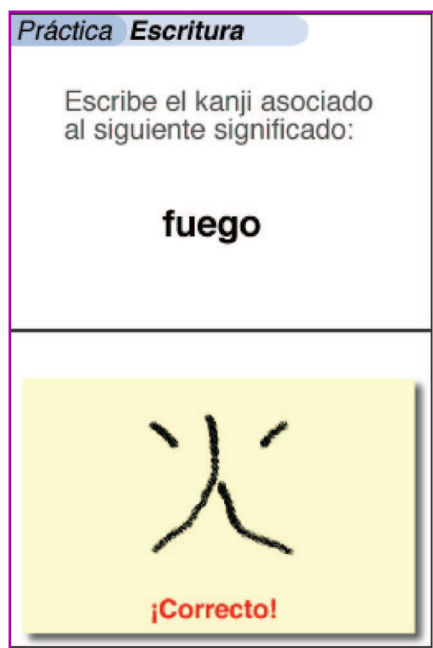


Figura 3: Juego de ordenación de numerales de KanaDS.

seleccionarlos de menor a mayor. Con este ejercicio se aprende rápidamente a relacionar los símbolos con el número que representan. Un juego de nivel más avanzado pediría al jugador que escribiese una cifra concreta escribiendo él mismo los kanji con la stylus en la pantalla.

De todas formas, hay que recalcar que Kana DS no pretende ser una solución total para aprender Japonés, sino un apoyo que complementa la etapa inicial de su estudio.

Planes de futuro

Desgraciadamente, para usar Kana DS será necesario adquirir un flash-card, que conlleva un coste considerable. Además, dependiendo del modelo, introducir el programa no suele ser tan sencillo como comprar un juego en la tienda e insertarlo en la consola.

La única solución para llevar el juego a todo el mundo sería realizar una versión comercial y presumiblemente de código cerrado y eso es algo que de momento está fuera de mi alcance. Pero quizás, en un futuro ...

El sitio de este proyecto está en [3].

OpenPipe

Por Santiago J. Barro Torres

Software Libre para los músicos del siglo que viene

El Proyecto

OpenPipe es un proyecto relacionado con la música, que nace con el objetivo de ser un punto de referencia dentro del mundo Open Source en su ámbito. Por una parte, es un proyecto de hardware libre, ya que abarca el

diseño de un instrumento musical digital desde cero (técnicamente, un controlador MIDI). Pero por otro lado, también es un proyecto de software libre, puesto que incluye el diseño de aplicaciones que permitan personalizar la OpenPipe.

El uso de la OpenPipe está pensado para aquellos músicos que deseen realizar actividades propias de su dominio (componer, editar partituras, ensayar melodías, etc.) utilizando para tal fin los medios que pone a su disposición la tecnología MIDI, la más empleada actualmente.

¿Qué es MIDI?

MIDI es un estándar que define cómo representar la información musical para su intercambio entre dispositivos electrónicos musicales: controladores MIDI, secuenciadores, sintetizadores, ordenadores, etc.

MIDI no representa el sonido directamente (como sucede con el mp3, por ejemplo), sino que almacena los eventos que permiten obtener ese sonido: *NOTA_ON(notas)*, *NOTA_OFF(notas)*, *VOLUMEN(notas, cantidad)*, etc. Se desacopla la información necesaria para tocar un sonido del propio sonido en sí. Fijémonos en los dispositivos MIDI representados en la Figura 4.

- **Controlador MIDI:** Es un aparato electrónico que imita la forma de un instrumento musical. Al mismo tiempo que el músico toca, se generan los eventos MIDI correspondientes a la interpretación.
- **Sintetizador MIDI:** Genera sonido a partir de una serie de secuencias MIDI. Lo habitual es que los sonidos estén pregrabados a partir de instrumentos reales.

Como se puede observar, la principal ventaja de la tecnología MIDI es su versatilidad. Un músico puede tocar con un controlador MIDI y escoger los sonidos del instrumento que prefiera. O también puede guardar los eventos generados durante su interpretación para editarlos posteriormente, por poner un par de ejemplos.

Proyecto OpenPipe

El objetivo del proyecto es crear un controlador MIDI con apariencia similar a la de una

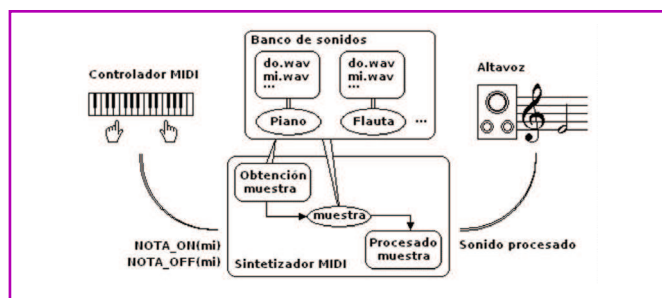


Figura 4: Aprende a escribir los kanjis con KanaDS.

flauta o una gaita. De tal forma, un músico habituado a tocar dichos instrumentos de viento (u otros similares: clarinete, oboe...) podrá utilizar la OpenPipe para cualquier actividad musical de las mencionadas anteriormente.

La OpenPipe consta principalmente de tres partes:

- Un cilindro de unos diez centímetros de altura y unos dos centímetros de diámetro, con ocho “sensores”, que el músico tapaná / destapaná para generar una nota, más un “sensor” de control adicional.
- Una pequeña unidad de procesamiento, en la que un microcontrolador se encargará de traducir la información obtenida a través de los “sensores” en información MIDI.
- Cableado de interconexión.

Ver la Figura 5.

La función básica de la OpenPipe es la de “capturar” la interpretación del músico, y convertir dicha interpretación en información MIDI. Sin embargo, la OpenPipe no genera sonidos directamente, sino que es preciso conectarla a algún tipo de sintetizador externo, utilizando para ello la salida MIDI proporcionada.

En caso de querer conectar la OpenPipe a un ordenador, es necesario un conversor MIDI-USB que puede adquirirse en cualquier tienda de música especializada.

Dentro del software libre podemos encontrar con varios sintetizadores software. Una de las aplicaciones más utilizadas es *timidity*, que admite diversos modos de funcionamiento. Sin duda, el más interesante de todos ellos es como servidor integrado con ALSA, que habilita la reproducción MIDI para cualquier aplicación del sistema.

Bancos de sonidos personalizados

Antes explicamos que un sintetizador es capaz de generar sonido a partir de eventos MIDI. Internamente, el sintetizador almacena un cierto número de muestras de cada instrumento que es capaz de generar. Por ejemplo,

treinta notas grabadas de un piano, otras treinta de una flauta, y así sucesivamente.

Supongamos que alguien quiere crear un instrumento personalizado con sus propias grabaciones. En concreto, pense-

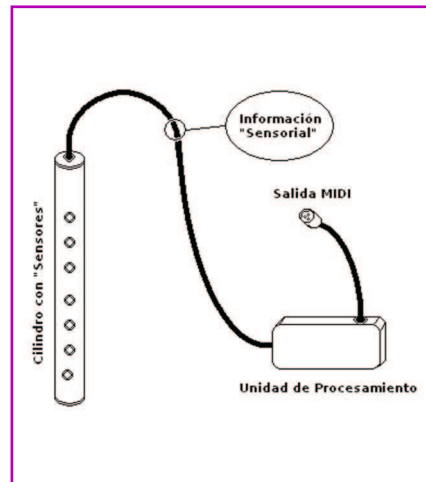


Figura 5: Esquema conceptual de un Sistema MIDI.

mos en un saxofonista que quiere utilizar la OpenPipe con los sonidos que ha grabado de su saxofón. Nos preguntamos lo siguiente: ¿Sería posible hacerlo? Afortunadamente, la respuesta es afirmativa.

Como el sintetizador es quien genera el sonido, está claro que es el mismo sintetizador quien debe proporcionar dicha funcionalidad. Aunque no haya un estándar al respecto, existen algunos formatos de intercambio de bancos de sonidos definidos por varios fabricantes. Por ejemplo, el formato SoundFont, cuyas especificaciones han sido publicadas por Creative Technology, es uno de los más utilizados. Por tanto, para utilizar bancos de sonidos personalizados, bastaría con diseñar un software que fuese capaz de crear ficheros conforme a las especificaciones SoundFont. Por supuesto, el sintetizador tendría que ser SoundFont-compatible (*timidity* lo es) para poder cargar el instrumento que hayamos creado.

Es interesante indicar que el proyecto OpenPipe dispondrá de un par de utilidades para crear bancos de sonidos en el mencionado formato.

En el blog del proyecto [4] podrás encontrar más información, incluyendo un enlace a la forja en la que se publicarán los diseños. ■

RECURSOS

[1] Información del desarrollo de CoolTran en la Forja-RedIRIS: <https://forja.rediris.es/projects/csl-colltran>

[2] Blog de CoolTran: <http://cooltran.wordpress.com>

[3] Sitio del proyecto KanaDS: <http://kanads.blogspot.com>

[4] Blog del proyecto OpenPipe: <http://openpipe.wordpress.com/>