

PROGRAMACIÓN DE VIDEOJUEGOS CON SDL

Programación de Videojuegos con SDL

Alberto García Serrano

Esta obra está bajo una licencia Attribution-NonCommercial-NoDerivs 2.5 de Creative Commons. Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-nc-nd/2.5/> o envíe una carta a Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

Índice.

| | |
|--|------------|
| Introducción | 4 |
| Historia de los videojuegos | 6 |
| La llegada del ordenador personal | 6 |
| Sistemas Operativos o la torre de Babel..... | 8 |
| Anatomía de un Videojuego | 9 |
| Entrada..... | 9 |
| Visualización..... | 9 |
| Sonido..... | 10 |
| Comunicaciones..... | 10 |
| Game loop..... | 10 |
| Los cimientos de un videojuego | 11 |
| La idea | 12 |
| Los elementos del juego..... | 12 |
| Manos a la obra..... | 14 |
| Primeros pasos con SDL | 23 |
| Trabajando con SDL..... | 24 |
| Video | 26 |
| Cargando y mostrando gráficos..... | 28 |
| Efectos especiales: Transparencias y alpha-blending..... | 32 |
| Otras funciones de interés..... | 36 |
| Gestión de eventos | 38 |
| Tipos de eventos | 39 |
| Lectura de eventos | 39 |
| Eventos del teclado | 40 |
| Eventos de ratón..... | 44 |
| Eventos del joystick..... | 45 |
| Otros eventos | 45 |
| Joystick | 47 |
| Recopilando información sobre el joystick | 47 |
| Leyendo el joystick..... | 48 |
| Audio | 51 |
| CD-ROM | 56 |
| El Window Manager | 60 |
| Timing | 60 |
| Librerías auxiliares para SDL | 62 |
| SDL_ttf..... | 62 |
| SDL_image..... | 66 |
| SDL_mixer..... | 69 |
| Sonidos..... | 70 |
| Música..... | 73 |
| Sprites: héroes y villanos | 76 |
| Control de sprites..... | 77 |
| Implementando los sprites..... | 79 |
| Utilizando nuestra librería..... | 86 |
| Un Universo dentro de tu ordenador | 91 |
| Almacenando y mostrando tiles..... | 93 |
| Diseñando el mapa..... | 98 |
| Scrolling..... | 100 |
| Enemigos, disparos y explosiones | 105 |

| | |
|---|------------|
| Tipos de inteligencia | 105 |
| Comportamientos y máquinas de estado..... | 107 |
| Disparos y explosiones | 108 |
| <i>¡Que comience el juego!</i> | 117 |
| Enemigos..... | 117 |
| Niveles..... | 119 |
| Temporización..... | 120 |
| Pantalla inicial, puntuación y vidas | 121 |
| ¿Y ahora que? | 136 |
| <i>Instalación de SDL</i> | 137 |
| Windows (VC++) | 137 |
| Linux..... | 141 |
| <i>De C a C++</i> | 143 |
| Clases y objetos | 144 |
| Herencia..... | 147 |
| Polimorfismo | 147 |
| Punteros y memoria | 148 |
| <i>Recursos</i> | 150 |
| Bibliografía | 150 |
| Programación | 150 |
| Programación de videojuegos | 150 |
| Gráficos | 151 |
| Enlaces | 151 |
| Programación | 151 |
| Programación de videojuegos | 151 |
| Gráficos | 152 |

Introducción

La imaginación es el motor de la creatividad, y la creatividad el motor de la inteligencia.

Muchas son las personas que opinan que la programación de ordenadores es una ciencia, otros lo ven como una ingeniería, sin embargo, algunos preferimos verla como un arte. Si tienes este libro entre tus manos, es porque te interesa la programación y también los videojuegos. Si es así, enhorabuena, porque estás a punto de entrar en un mundo realmente fascinante. La programación de videojuegos es muy diferente a la programación tradicional, requiere de una gran creatividad, pero también de una gran curiosidad y ganas de descubrir y, por que no, inventar cosas nuevas. De hecho, cuando creamos un videojuego, estamos creando un mundo nuevo, con su propio aspecto y sus propias leyes. En ese sentido el programador es como un dios dentro de ese mundo. La mayoría de las personas que se dedican a la programación se han planteado en algún momento el hacer un juego, y sin embargo son pocos los programadores que se deciden a hacerlo. ¿Hace falta algún don especial para hacer videojuegos? Probablemente no, pero hay muy buenos escritores de novelas que no son capaces de escribir poesía, y viceversa. En programación parece ocurrir lo mismo, hay programadores especialmente dotados para la programación de gestión y otros se desenvuelven mejor con una programación más cercana a la máquina.

Supondré que el lector ya posee conocimientos de programación, y más concretamente del lenguaje C. En el transcurso del libro, vamos a introducir las técnicas básicas utilizadas en la creación de videojuegos de forma práctica y a través de ejemplos, incluyendo el desarrollo de un juego completo. Vamos a utilizar para ello una librería multiplataforma llamada SDL (Simple Directmedia Layer) que nos va a permitir que nuestros juegos sean portables entre Windows, Linux, Macintosh, BeOs y otros. También vamos a introducir ciertas características de la programación orientada a objetos (POO) que nos ofrece C++, pero no debe preocuparse el lector desconocedor de la POO, porque dedicamos un apéndice completo para familiarizarnos con los elementos que vamos a usar de dicho lenguaje. La programación orientada a objetos es hoy necesaria a la hora de conseguir juegos de calidad profesional, por ello, introduciremos ciertos aspectos de este paradigma de programación.

Respecto a al grafismo utilizado en el juego de ejemplo que desarrollaremos durante el libro, he utilizado la librería de sprites GPL de Ari Feldman (<http://www.arifeldman.com/>), al cual agradezco, y animo a seguir ampliando.

Antes de acabar, quiero hacer una aclaración. Este no es un libro sobre programación y uso de SDL. Es un libro sobre programación de videojuegos, por lo tanto, no esperes una exhaustiva referencia sobre SDL. Iremos descubriendo los diferentes aspectos de esta librería según los necesitemos en nuestro camino a la creación de un videojuego, es por ello que habrá elementos de la librería que no formarán parte de los temas cubiertos por este libro.

Quiero, finalmente, dejar patente el objetivo que siempre he perseguido mientras escribía el presente texto: Mantener las cosas simples, sin entrar en tecnicismos innecesarios que distraigan al programador menos formal del objetivo final, que es dotarle de las herramientas necesarias para expresar toda su creatividad en forma de videojuego. No me cabe duda que hay cientos o miles de potenciales buenos programadores de juegos que, aún teniendo los conocimientos de programación necesarios, no han podido acceder a una información clara y carente de formalismos más

propios de un texto universitario. Espero que este sea un libro “para todos”, aunque ello signifique sacrificar en precisión y formalidad. El intentar trasladar la información contenida en el libro a un lenguaje sencillo y accesible me ha resultado a veces ciertamente una difícil tarea, por lo que pido excusas si en algún momento no he conseguido expresar las ideas con la suficiente simplicidad y claridad.

Puede contactar con el autor en la dirección de correo electrónico sdlintro@agserrano.com.

Historia de los videojuegos

Allá donde el tiempo pierde su memoria, grandes hombres y mujeres construyeron nuestro futuro.

La prehistoria de los videojuegos, se remonta a 1958. Bill Nighinbothan presenta en una feria científica un aparato que permite, mediante unos potenciómetros, mover una pequeña raqueta en un tubo de rayos catódicos. Bill no fue capaz de intuir el vasto potencial que aquel pequeño aparato tenía. Basándose en este aparato, Nolan Bushnell crea en 1972 un videojuego llamado *Pong*. Nollan, que sí es capaz de ver las posibilidades de este nuevo mercado, funda Atari, pero antes, Nollan ya había comercializado su primer videojuego: *Computer Space*. Años más tarde, en 1976, un empleado de Atari llamado Steve Jobs, ayudado por Steve Wozniak, crean un videojuego llamado *BreakOut*. Dos años después, ambos dejarán Atari para crear Apple Computer y pasar a la historia de la informática.



Steve Wozniak y Steve Jobs en su garaje, california. 1976

Es en 1978 cuando Taito lanza al mercado el famoso *Space Invaders*. Este juego era capaz de almacenar las puntuaciones máximas, convirtiéndose en todo un clásico.

En los siguientes años, comienzan a aparecer en el mercado nuevos videojuegos de excelente calidad, y que se convertirán en clásicos. Juegos como *Donkey Kong*, *Frogger*, *Galaga*, *Pac Man*, etc...

La llegada del ordenador personal

En la década de los 80, los ordenadores personales comienzan a popularizarse, y ponen en manos de particulares la posibilidad, no sólo de jugar a videojuegos creados para estos ordenadores, sino que ahora cualquiera puede adentrarse en el mundo de la programación de

videojuegos. Estos pequeños ordenadores de 8 bits hicieron las delicias de miles de aficionados a la informática. En España se hizo muy popular el Sinclair Spectrum. Un pequeño ordenador muy económico con un procesador Z80 y una memoria de 48 kilobytes. Otros ordenadores similares fueron el Amstrad 464, MSX y Commodore 64.



Commodore 64

De esta época tenemos juegos tan recordados como *Manic Miner*, *Xevious*, *Jet Set Willy*, *Arkanoid* y otros. A finales de los 80, surgen grupos de programadores en el Norte de Europa que comienzan a crear pequeñas joyas de la programación que intentaban exprimir el potencial de estas pequeñas máquinas al máximo. Concretamente estos grupos se crean en torno a la máquina de Commodore, gracias a sus posibilidades gráficas y de sonido, algo más avanzadas que la de sus competidores de la época. Estos pequeños programas es lo que hoy se ha dado en llamar *Demos*. Son por tanto los creadores de la actual *demoscene*.

En la década siguiente, aparecen ordenadores más potentes. Son equipos con microprocesadores de 16 bits y tienen entre 512 kbytes y 1 Mb de memoria. Quizás el más popular de estos fue el Amiga 500, también de Commodore, aunque sin olvidar al Atari ST o al Acorn Archimedes. Estos ordenadores tenían unas capacidades gráficas muy superiores a sus antecesores. Podían utilizar 4096 colores simultáneos, y su capacidad de sonido también comenzaba a ser bastante espectacular. De esta época son juegos como *Defender of the Crown*, *Barbarian*, *Elvira*, *Lemings* y *Maniac Mansion*.

Paralelamente a esta evolución, otro ordenador iba ganando adeptos a marchas forzadas. Sus capacidades gráficas y de sonido no eran mejores que la de sus competidores, de hecho eran bastante inferiores, sin embargo, lograron hacerse poco a poco con el mercado de los ordenadores personales, hasta el punto de reinar casi en solitario (con permiso del Apple Macintosh). Evidentemente hablamos del PC. Afortunadamente, al convertirse en el sistema mayoritario, sus capacidades gráficas y de sonido comienzan a avanzar a pasos agigantados. Una fecha clave es abril de 1987, cuando IBM presenta en el mercado la tarjeta gráfica VGA. Dejando obsoletas a las legendarias EGA y CGA. A la VGA la seguiría la SVGA, y después toda una pléyade de tarjetas con cada vez más memoria y aceleración gráfica 3D, llegando hasta las actuales Voodoo, las Gforce de Nvidia y las Radeon de ATI, con unas capacidades gráficas fuera de lo común a un precio muy asequible. Como es normal, esta época está marcada por juegos como *Quake*, *Counter-Strike (half life)*, *Heretic*, *Warcraft* y *Hexen*.

Sistemas Operativos o la torre de Babel

El ordenador PC, comercializado por IBM en el año 1982, ha pasado por varias etapas de existencia. Aquellos primeros PCs iban equipados con el sistema operativo PC-DOS, que no era más que una versión de IBM del MS-DOS de Microsoft. En los tiempos del MS-DOS la programación de videojuegos era bastante compleja, hacían falta grandes conocimientos de lenguaje ensamblador y de la estructura interna de la máquina, así como de la tarjeta de video y de sonido. Toda la programación se hacía directamente sobre el hardware. Es fácil imaginar los inconvenientes, había que preparar el juego para que funcionara con muchos modelos de tarjeta gráfica y también para los diferentes modelos de tarjetas de sonido (Adlib, Gravis Ultrasound (GUS), Sound Blaster). Es decir, hacer un juego, por simple que éste fuera, requería un esfuerzo muy alto, sin contar con que no siempre toda la información necesaria para acceder al hardware estaba al alcance de todo el mundo.

La llegada de Windows tampoco mejora demasiado el panorama, ya que no permite un acceso directo al hardware y su rendimiento gráfico es pobre. El acceso a los recursos del sistema también varía de un Sistema Operativo a otro. Se hace imprescindible, pues, crear una interfaz común para el desarrollo de videojuegos que permita el acceso estandarizado y con una interfaz común a los recursos del sistema. Algunas de estas propuestas son OpenGL y DirectX (éste último sólo para sistemas Windows).

En el caso de OpenGL, se nos ofrece una interfaz multiplataforma, sin embargo, se limita a ofrecernos una interfaz de acceso a gráficos (especializada en 3D, aunque con posibilidades de 2D) y no nos permite el acceso a otros recursos del sistema (teclado, sonido, joystick, timers, etc...).

DirectX, sin embargo, está compuesto por varias sub-interfaces con el SO y el hardware, como DirectDraw (Gráficos 2D), Direct3D (Gráficos 3D), DirectInput (Entrada y Salida), DirectPlay (Capacidades de comunicación en redes TCP/IP), DirectSound (Acceso a hardware de sonido) y algún que otro componente más. Como vemos, DirectX tiene gran cantidad de capacidades, pero presenta dos inconvenientes. El primero es que sólo es válido para sistemas Windows, y el segundo es que por la propia naturaleza de la programación en Windows, su uso es algo engorroso y no apto para programadores con poca experiencia con las APIs de Windows y más concretamente con la interfaz COM+.

Afortunadamente, la empresa Loki Games, ante la necesidad de hacer portables los juegos de Windows a Linux y a otros sistemas operativos, crean una librería multiplataforma llamada SDL (Simple Directmedia Layer), que nos ofrece acceso a los recursos de la máquina (gráficos, sonido, entrada/salida, timing, etc...) mediante una interfaz coherente e independiente del SO. SDL, además, nos permite utilizar la interfaz OpenGL para gráficos 3D.

A square graphic with a light gray background. At the top, the word "Capítulo" is written in a bold, black, sans-serif font. Below it, a large, white, stylized number "2" is centered.

Anatomía de un Videojuego

Somos nuestra memoria, somos ese quimérico museo de formas inconstantes, ese montón de espejos rotos.

Jorge Luis Borges

Cada vez que juegas a tu videojuego preferido, dentro del ordenador están ocurriendo muchas cosas. Se hace patente que un factor importante en un juego es que se mueva con soltura y a buena velocidad. Todos los elementos del juego parecen funcionar de forma independiente, como con vida propia y a la vez. Sin embargo esto es sólo una apariencia, ya que dentro del programa se van sucediendo las diferentes fases de ejecución de forma secuencial y ordenada. En este capítulo vamos a tratar de dar una visión general y sin entrar en detalles de implementación de la anatomía de un videojuego. Vamos a ver qué partes lo componen y como se relacionan.

Entrada.

Un videojuego necesita comunicarse con el jugador a través de un dispositivo de entrada, como un Joystick o un teclado. La programación del sistema de Entrada/Salida puede parecer a priori algo fácil de implementar, y de hecho lo es. Sin embargo, si no queremos ver como nuestro estupendo juego se hace poco "jugable" para el jugador hay que cuidar mucho este aspecto. Una mala respuesta del juego ante una pulsación del teclado puede hacer frustrante la experiencia de juego para el jugador. Otro factor importante es el soporte de una variedad de dispositivos de entrada. Algunos jugadores preferirán usar el teclado al ratón y viceversa. Si además queremos que nuestro juego tenga un toque profesional, hay que dar la posibilidad al jugador de que defina las teclas con las que quiere jugar, además de dar soporte a los diferentes tipos de teclados internacionales.

Visualización.

La misión de esta capa es la de convertir el estado interno del juego en una salida gráfica. Ya sea en 2D o 3D, la calidad gráfica es un factor importantísimo. Normalmente, el programador trabaja con unos gráficos creados por él mismo, y cuya calidad dependerá de las dotes artísticas de éste. En una siguiente fase, un artista gráfico crea y sustituye estas imágenes por las definitivas. Uno de los elementos principales de un buen juego es un buen motor gráfico, capaz de mover una gran cantidad de Sprites en un juego 2D o una gran cantidad de polígonos (triángulos) en el caso de un juego en 3D. En el capítulo 6 veremos qué es un Sprite.

Gracias por visitar este Libro Electrónico

Puedes leer la versión completa de este libro electrónico en diferentes formatos:

- HTML(Gratis / Disponible a todos los usuarios)
- PDF / TXT(Disponible a miembros V.I.P. Los miembros con una membresía básica pueden acceder hasta 5 libros electrónicos en formato PDF/TXT durante el mes.)
- Epub y Mobipocket (Exclusivos para miembros V.I.P.)

Para descargar este libro completo, tan solo seleccione el formato deseado, abajo:

