

Tema 7: Aspectos Avanzados de SDL - La Librería TTF

Tecnología de la Programación de Videojuegos 1
Grado en Desarrollo de Videojuegos

Miguel Gómez-Zamalloa Gil

Departamento de Sistemas Informáticos y Computación
Universidad Complutense de Madrid

La Librería TTF

- ✦ Permite utilizar fuentes **TrueType** en aplicaciones SDL
- ✦ **TrueType** es un formato estándar de tipos de letra escalables
 - ▶ Creado por Apple y Microsoft en los años 80
 - ▶ Ficheros con extensión **.ttf**
 - ▶ Elementos vectoriales que emplean funciones cuadráticas
 - ▶ Las funciones cuadráticas son convertidas a mapas de bits teniendo en cuenta el tamaño y otros aspectos
- ✦ Otros formatos de fuentes más avanzados:
 - ▶ **OpenType (OTF)**: Creado por Adobe y Microsoft
 - ▶ **Apple Advanced Typography (AAT)**: Extiende TTF con características similares a las de OTF

La Librería TTF - Instalación

- ✦ Es una librería extra y requiere instalación (diferente dependiendo del entorno)
- ✦ En nuestra plantilla `ProyectosSDL`:
 - ▶ Copiar la librería en el directorio, modificar/actualizar el archivo de propiedades y copiar las `dlls` en el directorio `bin` (ver las instrucciones del fichero `README`)
- ✦ Para usarla hay que incluirla:

```
#include <SDL_ttf.h> // Puede variar dependiendo de la instalación
```

- ✦ Las funciones, tipos y constantes empiezan por `TTF_`

La Librería TTF - Funciones Básicas

1. Inicialización

```
TTF_Init();
```

2. Carga de una fuente

```
TTF_Font* font;  
font = TTF_OpenFont(filename.c_str(),size); // Fichero .ttf y tamaño en ptos
```

3. Generación de textura a partir de la fuente y un texto

```
SDL_Surface* surf = TTF_RenderText_Solid(font, text.c_str(), color);  
SDL_Texture* texture = SDL_CreateTextureFromSurface(renderer, surf);
```

4. Renderización (como cualquier otra textura)

```
SDL_RenderCopy(renderer, texture, &srcRect, &destRect);
```

5. Liberación de la fuente

```
TTF_CloseFont(font);
```

6. Finalización

```
TTF_Quit();
```

TTF - Integración en nuestro Framework

♦ Creamos una nueva clase Font

```
class Font {  
    private:  
        TTF_Font* font = nullptr;  
    public:  
        Font() {};  
        Font(string filename, int size) {  
            load(filename, size);  
        }  
        ~Font() { free(); }  
        bool load(string filename, int size) {  
            font = TTF_OpenFont(filename.c_str(), size);  
            ...  
        }  
        void free() {  
            if (font != nullptr) TTF_CloseFont(font);  
            font = nullptr;  
        }  
        SDL_Surface* generateSurface(string text, SDL_Color color) const {  
            return TTF_RenderText_Solid(font, text.c_str(), color);  
        }  
};
```

TTF - Integración en nuestro Framework

- ♦ Añadimos un nuevo método de carga en la clase **Texture**

```
void Texture::loadFromText(SDL_Renderer* renderer, string text, const Font& font,
                           SDL_Color color) {
    SDL_Surface* textSurface = font.generateSurface(text, color);
    if (textSurface == nullptr)
        throw exception("Unable to render text surface: " + TTF_GetError());
    else {
        free();
        texture = SDL_CreateTextureFromSurface(renderer, textSurface);
        if (texture == nullptr) {
            throw exception("Unable to create texture from text: " + SDL_GetError());
            w = h = 0;
        } else {
            w = textSurface->w;
            h = textSurface->h;
        }
        SDL_FreeSurface(textSurface);
    }
}
```

La Librería TTF - Más Funciones

♦ A parte de `TTF_RenderText_Solid`, la librería dispone de otras 11 funciones para el renderizado de texto

- ▶ Tres tipos de renderizado: **Solid, Shaded, Blended**
- ▶ Cuatro formatos de texto: **Glyph, Text, UTF8 y UNICODE**
- ▶ Las 12 funciones corresponden con todas las posibles combinaciones:

```
TTF_RenderFormatoTexto_TipoRenderizado(TTF_Font* font, ... text,  
                                         SDL_Color color)
```

♦ Las fuentes tienen además una serie de propiedades y atributos que se pueden establecer y obtener.

- ▶ Por ejemplo, el estilo (normal, cursiva, negrita, subrayado):

```
TTF_SetFontStyle(font, TTF_STYLE_BOLD | TTF_STYLE_UNDERLINE);
```

♦ Ver <http://www.sdltutorials.com/sdl-ttf> para más detalles