

Propuesta de Trabajo Final

Carrera de Ingeniería de Sistemas

Tema: Un aporte al desarrollo de aplicaciones interactivas en el Sistema Argentino de TV Digital Terrestre (SATVD-T)

Alumnos:

María Fernanda Ochoa

Daniel Néstor Uranga

Director: Dr. Cristian García Bauza

Codirector: Ing. Enzo Ferrante

Introducción:

En los últimos años, la implementación del Sistema Argentino de TV Digital Terrestre ha generado un nuevo abanico de posibilidades para el desarrollo de aplicaciones interactivas. La norma adoptada para estandarizar dicho sistema, ISDB-Tb, plantea la utilización de los lenguajes de programación NCL/Lua y del middleware Ginga [1] como capa de abstracción sobre el hardware subyacente de cada Set Top Box (STB), posibilitando de este modo el desarrollo de software capaz de ser ejecutado independientemente de los proveedores del equipo. Ginga ha sido adaptado para su implementación en Argentina por el LIFIA (Laboratorio de Investigación y Formación en Informática Avanzada, Facultad de Informática, Universidad Nacional de La Plata) y rebautizado como Ginga.ar.

La creación de aplicaciones interactivas en general y de videojuegos en particular, constituye un trabajo interdisciplinario en el que interactúan profesionales de diferentes áreas del conocimiento, desde los artistas creadores de la gráfica hasta los programadores artífices del diseño e implementación de la aplicación.

Así como los diseñadores gráficos utilizan herramientas de software para realizar sus creaciones, los informáticos requieren hacer uso de un conjunto de piezas de software que les permitan desarrollar sus productos de manera eficiente. Una de éstas, son las denominadas Bibliotecas de Desarrollo, que brindan a los programadores un repertorio de artefactos capaces de simplificar la tarea y organizar la estructura de la aplicación a ser desarrollada. Según el tipo de aplicaciones para el cual fue desarrollada, la biblioteca deberá cubrir un conjunto de funcionalidades básicas. Por ejemplo, las bibliotecas para el desarrollo de aplicaciones interactivas y videojuegos, en general brindan soporte para manejo de sprites,

composición de escenas, animaciones, mapas basados en tiles, detección de colisiones y otros mecanismos para facilitar las tareas comunes al desarrollo de cualquier videojuego 2D.

Antecedentes:

A lo largo de los últimos años, gracias al advenimiento de nuevas tecnologías de cómputo y procesamiento gráfico, el uso de videojuegos y aplicaciones interactivas en las computadoras modernas ha sufrido un drástico incremento. Este fenómeno protagonizado por los usuarios, ha tenido su contraparte del lado de los ingenieros y programadores de software, quienes para lidiar con el dinamismo de este mercado y mejorar la calidad y el tiempo invertido en el proceso de desarrollo, han creado distintas bibliotecas para simplificar esta tarea. Actualmente existe una amplia gama de opciones enfocadas según el tipo de aplicación a desarrollar.

Para el desarrollo de videojuegos en 2D, una de las más destacadas es la biblioteca abierta y multiplataforma **SDL** (*Simple Direct Media Layer*) [2] que provee una capa de abstracción para simplificar el trabajo al programador de videojuegos. Incluye facilidades para el manejo de audio, teclado, mouse, joystick, aceleración 3D para hardware (OpenGL) y manejo de Sprites en 2D. Por otro lado, el paquete **Allegro** (*Allegro low level game routines*) [3] se destaca por ser una biblioteca multiplataforma que simplifica el manejo de tareas de bajo nivel, tales como creación de ventanas, carga de datos, dibujo de imágenes, etc. **ClanLib** [4] permite interactuar con nuevas tecnologías tales como OpenGL3, renderizado basado en shaders, soporte para XML, entre otras. **PyGame**[5] es una biblioteca para desarrollo en Python que posee soporte para tecnologías actuales como procesamiento multicore.

Adicionalmente, existen bibliotecas que se han tornado populares debido al dominio de aplicación, por ejemplo **Cocos2d** para el desarrollo de videojuegos 2D focalizado en dispositivos móviles y **Ogre3D** para mundos tridimensionales, junto con **Unity** para videojuegos en PC de alto desempeño.

Uno de los lenguajes que ha ganado lugar en el ámbito del desarrollo de videojuegos simples es LUA[6]. Este lenguaje dinámico de *scripting*, ha sido elegido por varios motores de creación de videojuegos como punto de entrada para el programador, debido a su simplicidad y flexibilidad. LUA fue adoptado por parte del sistema argentino TVD-T como el lenguaje estándar, junto al lenguaje declarativo NCL[7], para escribir aplicaciones capaces de ser ejecutadas en los set Top Boxes hogareños.

El desarrollo de videojuegos para la plataforma Ginga.ar del sistema Argentino de TVD-T, es un campo que se encuentra en sus primeras etapas de desarrollo. Para las plataformas existentes de Ginga (Ginga-J o GingaNCL/LUA) hay distintas bibliotecas de desarrollo.

En el caso de **Ginga-J** existen dos bibliotecas desarrolladas en Brasil para la construcción de videojuegos. **Tuga** [8] es un middleware ideado para proveer a los desarrolladores de juegos en Ginga-J una capa de abstracción que simplifica el manejo de sprites, fuentes, personajes y

sonido; y **GingaGame** [9] es otro de los frameworks desarrollados con el objeto de acercar el desarrollo de juegos de TV Digital al desarrollo de juegos para PC. Al igual que Tuga, fue diseñado sobre Ginga-J y permite a los desarrolladores enfocarse en el código específico del juego, sin preocuparse demasiado por atender otras cuestiones básicas de implementación.

En el caso de **GingaNCL/LUA** existe sólo una biblioteca para el desarrollo de videojuegos denominado **ATHUS** [10], también de origen brasileño. Dicha biblioteca provee a los programadores de NCL/LUA un buen marco de referencia a la hora de implementar y diseñar videojuegos pero omite diversas funcionalidades y servicios. Actualmente, su desarrollo ha sido discontinuado, por lo que se torna necesario contar con un producto estable y en desarrollo continuo que cubra estas carencias.

Adicionalmente, existe una falencia real en cuanto a la creación de interfaces gráficas en el desarrollo de aplicaciones para TVD-T, en particular aquellas que requieran de diversos componentes (también conocidos como Widgets en inglés) como etiquetas, distintos tipos de botones, menús desplegables, etc.

Las necesidades expuestas anteriormente justifican ampliamente el objetivo de esta tesis, el cual se resume en la creación de una biblioteca para el desarrollo de aplicaciones interactivas y una biblioteca de widgets en el marco del Sistema Argentino de TV Digital Terrestre.

Objetivo:

El objetivo del trabajo es construir una biblioteca que facilite el desarrollo de aplicaciones interactivas en lenguaje NCL/Lua capaces de ser ejecutadas en la plataforma Ginga.ar soportada por los decodificadores del SATVD-T. Se pretende además implementar un conjunto de aplicaciones que den muestra de las capacidades de la biblioteca desarrollada y, a la vez, realicen un aporte a la resolución de problemáticas concretas haciendo uso esta nueva tecnología.

Plan de trabajo:

Las actividades propuestas para el presente Plan de Trabajo son:

1. Relevamiento de las capacidades brindadas actualmente por los decodificadores del SATVD-T.
2. Relevamiento del estado del arte en cuanto al desarrollo de aplicaciones para la plataforma Ginga.ar del Sistema Argentino de TV Digital Terrestre.
3. Análisis, diseño e implementación de una biblioteca que simplifique el desarrollo de aplicaciones para la plataforma Ginga.ar.
4. Análisis y definición de las aplicaciones a implementar haciendo uso de la biblioteca desarrollada.
5. Implementación de las aplicaciones definidas.

6. Redacción del proyecto Final.

Referencias:

- [1] Televisión digital terrestre - Guía de operación Parte 3: Multiplexación y servicio de información (SI) — Guía para implementación de la ABNT NBR 15603:2007
- [2] B. Pendleton, "Game programming with the sdl", Linux J., vol. 2003, pp. 1-, June 2003.
- [3] "<http://alleg.sourceforge.net/>," 2011. Allegro. A game programming library.
- [4] "http://clanlib.org," 2011. ClanLib Game SDK.
- [5] "http://pygame.org," 2011. Sitio web de PyGame.
- [6] Lua 5.1 Reference Manual. R. Ierusalimschy, L. H. de Figueiredo, W. Celes. 2006
- [7] Programando em NCL 3.0 2a. L. F. Gomes Soares, S. D. Junqueira Barbosa. 2011
- [8] D. A. Ferreira and C. T. Souza, "Tuga: Um middleware para o suporte ao desenvolvimento do jogos em tv digital interativa.," 2009. Disponible en: <http://code.google.com/p/tugasdk/>.
- [9] D. C. Barboza and E. W. G. Clua, "Ginga game: A framework for game development for the interactive digital television," in proceeding of the 2009 VIII Brazilian Symposium on Games and Digital Entertainment, SBGAMES '09, (Washington, DC, USA), pp. 162-167, IEEE Computer Society, 2009.
- [10] R. M. Segundo, J. C. F. da Silva, and T. A. Tavares, "Athus: A generic framework for game development on ginga minddleware," Game and Digital Entretainment, Brazilian Symposium on, vol. 0, pp. 89-96, 2010.