Interfaz grafica de usuario

Para la estructura definida de escena o show, la interfaz grafica cuenta con formularios sencillos de altas, bajas y modificaciones para todos los tipos de objetos existentes. Se diseñó en base a barras de herramientas flotantes de forma de poder tener visibilidad de la escena en todo momento y observar así cómo los cambios efectuados impactan sobre ella.

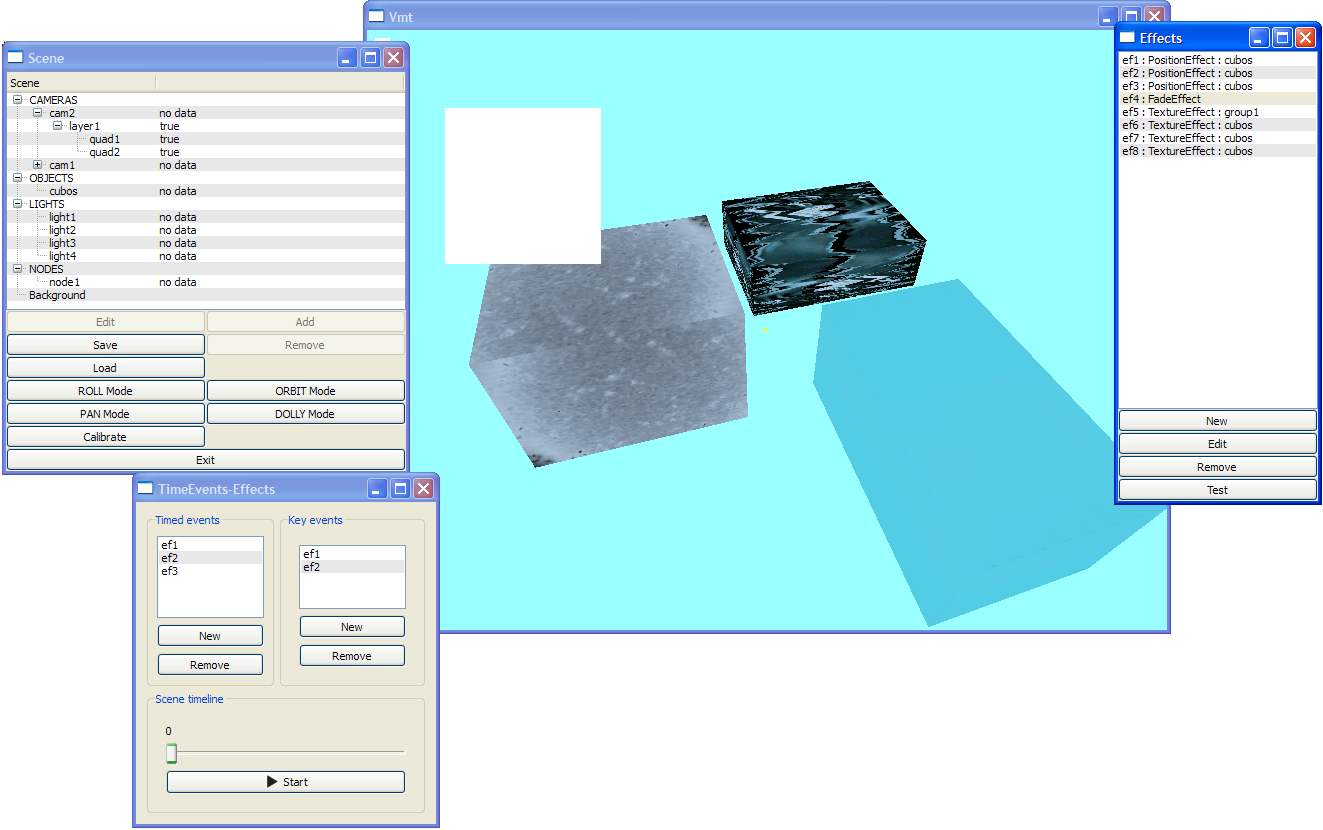


Figura: Vista general de la interfaz de usuario

Al inicio se cuenta con una vista grafica de la escena desde el punto de vista de la cámara activa, una vista de los objetos contenidos en la escena, la lista de definición de efectos y una línea de tiempo con las listas de efectos por tiempo y por evento.

En la ventana de escena se manejan los objetos bidimensionales y tridimensionales, cámaras, luces, capas y los nodos distribuidos del *show*. También es desde donde se ejecuta la calibración de las cámaras-proyectores. Para la visualización de la escena, se ofrecen diferentes modos de cámara para girar u orbitar alrededor del centro o moverse en cada uno de los ejes.

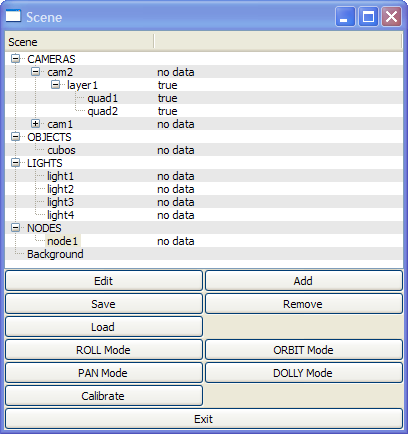


Figura: ventana para manejar la escena

* Cámaras, Capas y *Quads*

Las cámaras tienen todas las propiedades usuales que se pueden encontrar en paquetes de software de animación tridimensional como ser Posición, *Eye*, Up, FOV (*Field of View*), Aspecto, *Near, Far*, etc.

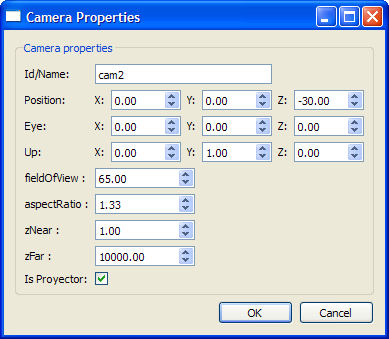


Figura: Dialogo de propiedades de Camara

Se han modelado los proyectores como cámaras con marcas especiales, por lo cual a cada cámara se le puede indicar además si de lo que se trata en realidad es de un proyector. Esto permite contar tanto con cámaras para las cuales lo que veamos será lo que efectivamente saldrá proyectado, como también poder tener cámaras sin proyector asociado que nos permiten tener una vista en perspectiva de la escena para por ejemplo, experimentar como se vería el *show* desde el punto de referencia de un cierto observador.

Existen también acciones para seleccionar los tipos de movimiento de la cámara activa. Estos son:

* *Roll*: posición y punto de vista fijos. Se rota alrededor del eje formado por ellos.
* *Orbit*: distancia al punto de vista fijo. Se orbita a dicha distancia alrededor del punto de vista.
* *Dolly*: punto y la dirección de vista fijos. Se mueve la posición de la cámara acercándose y alejándose del objetivo.
* *Pan*: distancia al plano de la imagen fija. Se mueve cualquier dirección pero en plano paralelo al de la dirección.

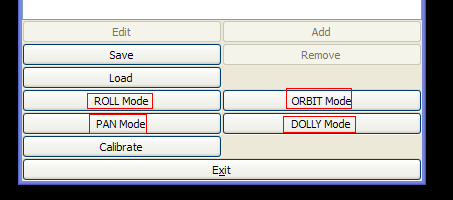


Figura: Acciones para seleccionar modo de movimiento de la cámara activa.

Una vez activado el modo de la cámara, manteniendo presionado el botón izquierdo y moviendo el mouse se mueve la cámara acorde al modo seleccionado.

Cada cámara es a su vez un contenedor de capas o *layers.* Su propósito es manejar todo lo que es mapeo sobre estructuras bidimensionales, y por ello son los contendores de *quads* bidimensionales sobre los cuales se pueden realizar todas las operaciones y efectos de mapeo disponibles.

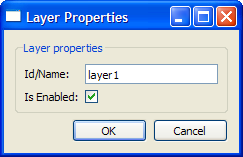
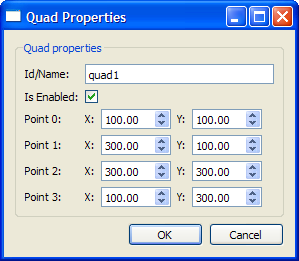
 

Figura: Propiedades de los objetos *Layer* y *Quad*

Ambos objetos tienen propiedades editables, en tiempo de edición o durante el *show*, para mostrarse u ocultarse completamente. Si se oculta una capa, todos los *quads* de la misma también serán ocultados. En caso de mostrarla, todos los *quads* visibles volverán a mostrarse.

* Objetos tridimensionales

Se permite agregar objetos tridimensionales con formato de malla triangular del tipo 3DS [ref]. Para satisfacer los requerimientos actualmente definidos, solo fue necesario permitir la modificación del identificador y la posición de un objeto tridimensional.

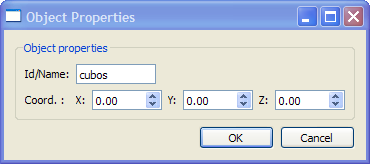


Figura: Dialogo de propiedades de objeto tridimensional

* Luces
* Efectos

Se proveen ventanas flotantes tanto para la creación de efectos de forma estática, esto es, la definición de qué es lo que hace o modifica el efecto en la escena, como para la asociación efecto-tiempo o efecto-evento en la línea de tiempo o en el show respectivamente. Si un efecto es definido para ser desplegado en un instante dado (efecto-tiempo) este se podrá visualizar cuando la línea de tiempo pase por ese instante dado. En caso de ser definido por evento (efecto-evento) este será visualizado cada vez que el evento asociado suceda.

Los tipos de efectos posibles son *de posición*, *degradé de color* y *mapeo de texturas*.

Los efectos de posición son aplicables a objetos tridimensionales solamente, y lo que permiten es animar el objeto moviéndolo de una posición inicial a una final. Es posible definir una demora en la ejecución del efecto comenzando en el instante en el que debería ser mostrado (al pasar la línea de tiempo por el instante definido si es un efecto asociado a tiempo o al reconocerse el código de evento del efecto si es lanzado por eventos).

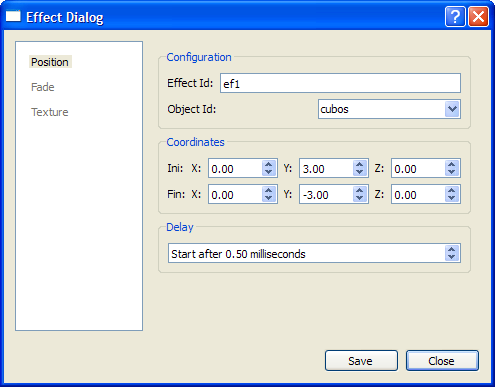


Figura: Dialogo de definición de efecto de tipo posición.

Los efectos de degradé solo son aplicables a grupos de *quads* y lo que permiten es pintar todos los *quads* del grupo de un color inicial e ir pasando por toda la gama de colores intermedios hasta llegar al color final. Para la selección de colores se utiliza un dialogo nativo para ese propósito. También es posible definir una demora al lanzar el efecto.

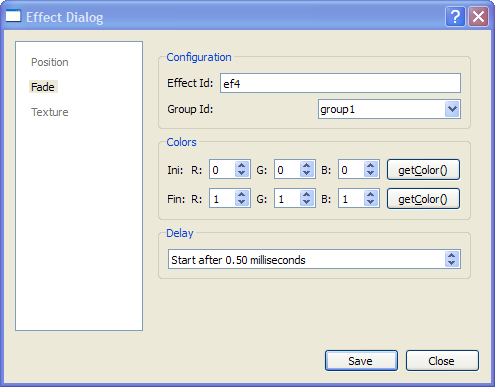


Figura: Dialogo de definición de efecto de tipo degradé o *fade*.

Los efectos de tipo textura son aplicables tanto a grupos de *quads* como a objetos tridimensionales. Se pueden definir texturas de tipo Imagen o Video. En ambos casos es necesario proporcionar el camino al archivo multimedia correspondiente. Para el caso especifico de un efecto de textura asociado a un objeto tridimensional, es necesario proporcionar además la cara o conjunto de caras (lista de selección de *faces*) sobre los cuales se va a mapear el efecto de tipo textura definido.

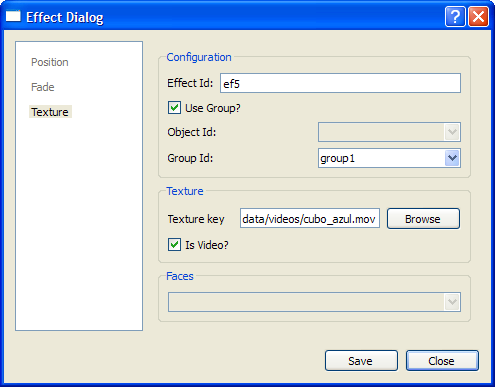


Figura: Dialogo de definición de efecto de tipo textura (en este caso aplicada a un grupo).

Si bien no hay un control explicito de la unicidad en los identificadores de los efectos, es recomendable no reutilizarlos. El resultado puede no ser el esperado en caso de existir más de un efecto con identificadores duplicados.

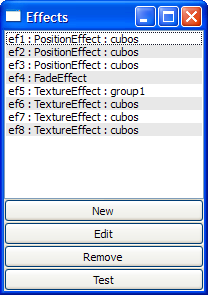


Figura: Lista de efectos definidos en la escena.

Es posible testear el efecto que se está creando presionando el botón *Test* de la lista de efectos. Esto forzara la ejecución por una única vez del efecto seleccionado en la lista.

* Calibración

Daniel rellenar.

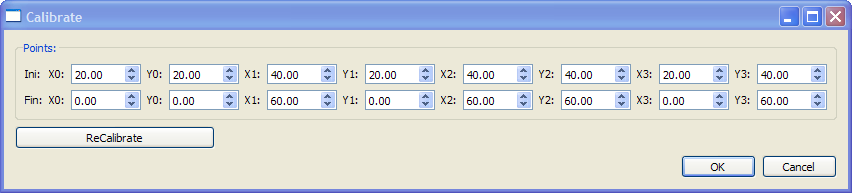


Figura: Dialogo para configurar y ejecutar la calibración.

* Línea de tiempo

Tanto para la fases de producción y revisión del evento como para la ejecución o reproducción misma del *show* en vivo, es necesario contar controles estándares para el manejo de la línea de tiempo.

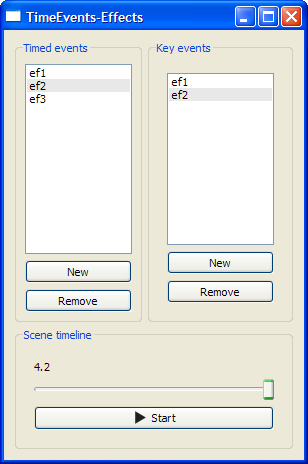


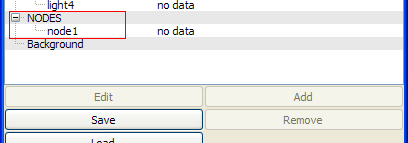
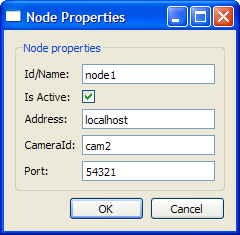
Figura: Diálogo con línea de tiempo y lista de efectos asociados a tiempo y eventos.

La aplicación provee una forma de iniciar el evento mediante la acción de *Start*. Se provee una vista grafica representada con una barra deslizante y una vista numérica para reflejar el avance en el tiempo en relación a la duración total del evento. Mediante la barra deslizante es posible navegar por la línea de tiempo e iniciarla visualización del evento desde un instante cualquiera. Esto último es muy útil para la fase de pre-producción del show en la cual es muy común que el usuario se quiera concentrar en una ventana de tiempo específica en la que se está trabajando.

Este dialogo también permite asociar los eventos previamente definidos tanto a un instante dado (lista *Timed events*) como a un código de evento (lista *Key events*) a ser utilizados tanto de teclado como dispositivos de entrada MIDI.

* Nodos remotos

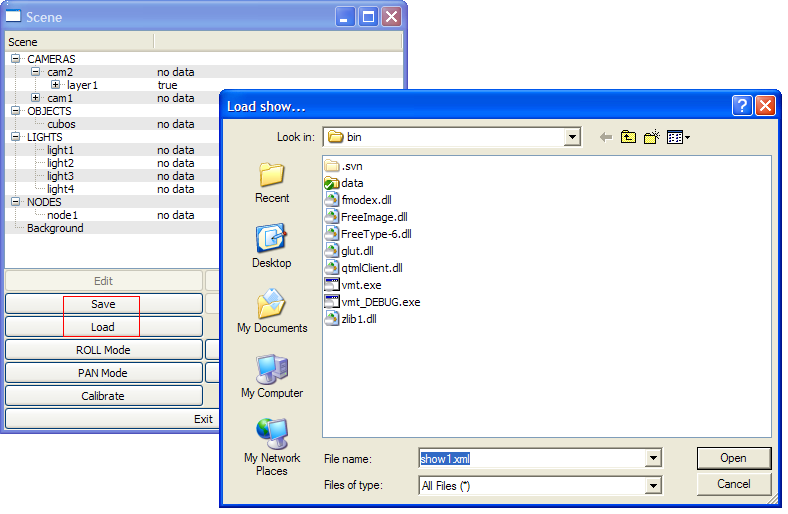
Para representar la red de nodos de VMT en los cuales se conectaran proyectores remotos, es preciso definir para cada nodo la IP y puerto donde estarán configurados y la cámara definida en la escena que estará asociada al proyector conectado a él.

Esta información es necesaria para establecer el canal de comunicación con el proceso remoto y para deducir que información se enviara al nodo en base a su cámara asociada. Recordar que las capas o *layers* están asociadas a una y solo una cámara, por lo cual si un efecto esta aplicado sobre un *quad* o grupo de *quads*, el mismo deberá ser propagado al nodo cuya cámara sea la que los contiene. En cuanto a los efectos aplicados sobre objetos tridimensionales, estos serán enviados a todos los nodos ya que todas las cámaras de todos los nodos tienen visibilidad de la escena tridimensional, aunque esta sea desde perspectivas diferentes.

* Cargar y Guardar proyecto XML

Se permite almacenar en archivos XML estándar una representación de todo lo que se ha definido en la escena. En el dialogo flotante de la escena se tienen el botón *Save* para persistir a archivo lo que se tiene definido en memoria hasta el momento y *Load* para cargar desde un archivo XML valido una escena previamente definida y almacenada.



Para la búsqueda de los archivos XML de escena se utilizan diálogos nativos para este propósito.

***Decisiones de implementación***

Para la implementación del modulo de interfaz grafica se siguió el patrón *Model-View-Controller*. Para cada ventana se generaron modelos que manejan los datos relevantes a cada una, impactando en el modelo general de la aplicación en caso de ser necesario.

Se utilizó *Qt Framework* [Qt - framework multiplataforma para la creación de aplicaciones e interfaces de usuario con APIs para C++. http://qt.nokia.com/products/] como biblioteca de base para la creación de las ventanas, sus controles gráficos y manejar tanto la comunicación que ocurre entre ventanas entre sí como con los modelos. También se hizo uso de la funcionalidad para internacionalización de las interfaces graficas, las que por el momento están tomando valores por defecto en idioma Inglés. Es multiplataforma por lo que puede ser utilizado en varios de los sistemas operativos de computadores de escritorio mas populares como ser Linux, Mac OS y Windows. Esto ultimo era un requerimiento fuerte del proyecto y fue tenido en cuenta al tomar la decisión de incorporar a la aplicación esta biblioteca para el desarrollo de la interfaz de usuario.