

Tutorial Ogitor

Tutorial Ogitor	1
1. Instalación	2
2. Crear un proyecto (nivel)	2
3. Interficie	3
4. Controles	5
5. Propiedades de la escena.....	7
6. Mundos paralelos.....	8
7. Añadir un Modelo	9
8. Añadir una luz.....	10
9. Añadir un sistema de partículas.....	13
10. Añadir un billboard set.....	14
11. Añadir una cámara	16
12. Game Objects	17

1. Instalación

Pasos a seguir:

- Descargar el instalador desde <http://www.ogitor.org/> apartado Downloads.
- Instalar con el .exe
- Abrir el programa, es el archivo (\$Directorio_Ogitor)/bin/qtOgitor.exe o con el acceso directo. Aseguraos que es la versión v0.4.0 beta.

2. Crear un proyecto (nivel)

- File->New o el icono correspondiente
- Aparecerá la ventana Project settings
- En *General* definiremos el Project Name y el Project Directory y dejaremos el resto de campos tal como están.
- En *Assets* con el botón derecho haremos Add directory o Add directory recursively y añadiremos el directorio y subdirectorios donde tengamos los ficheros .mesh, .material, las texturas de los modelos, etc... (*Importante:* Ogitor coge la posición relativa de los directorios de modo que si queremos cambiar el proyecto de directorio una vez creado habrá que repetir ese paso)
- En *Appearance* lo dejaremos como esta.
- Comprobamos que hemos creado el proyecto correctamente y que en la pestaña Meshes, abajo a la derecha, aparecen los modelos de los directorios definidos en el paso 4.

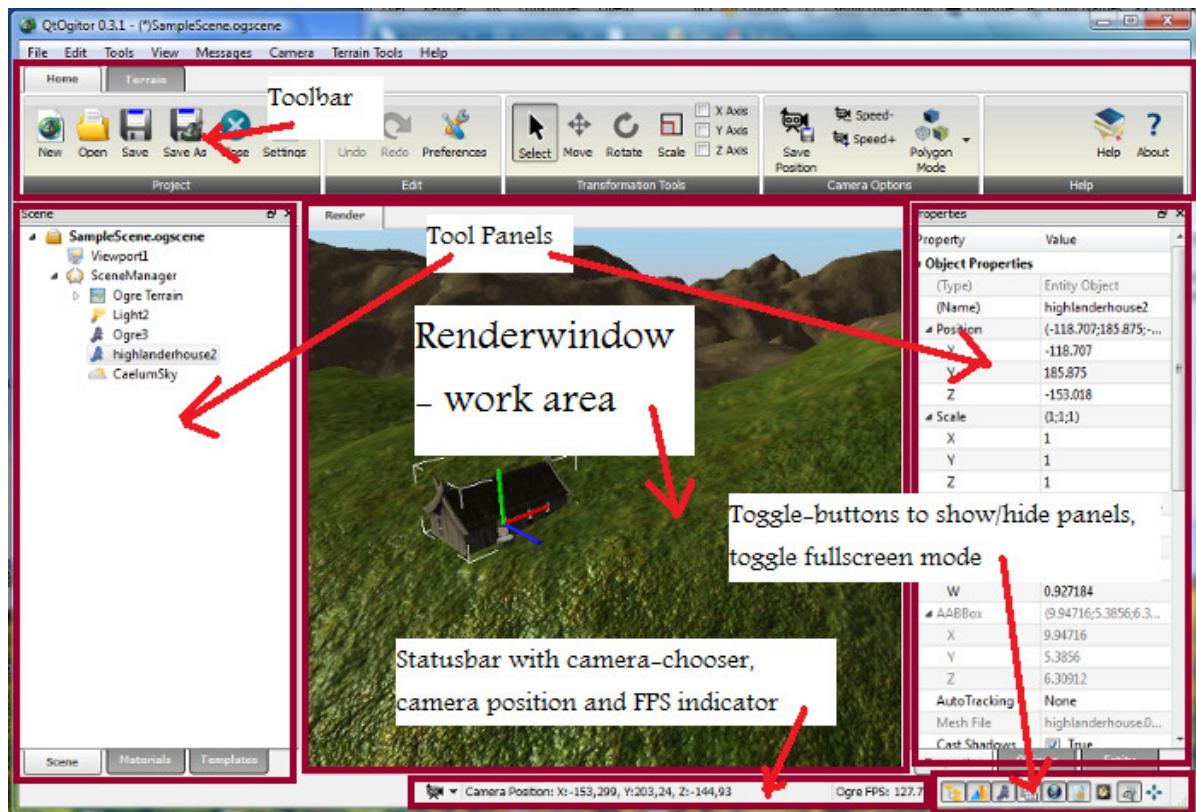
Para volver a acceder a la ventana Project settings podemos acceder desde edit->Settings.

3. Interficie

Texto de la documentación de Ogitor

The interface of qtOgitor is composed of five main elements:

- **Render window** - the main work area.
This is where the main interaction with Ogitor occurs.
- **Toolbar.**
The toolbar is where the most common operations are: scene commands, transformation tools, etc.
- **Tool panels.**
These dockable tool panels can be shown/hidden and moved around, either docked and floating.
 - **Scene Panel**
Hierarchical view of the scene. Select nodes in the tree and see their properties in the Property panel.
Selecting nodes in the tree will select them in the main view, and the other way around.
 - **Materials Panel**
A treeview of all available materials. Drag a material to an mesh object in the scene to change it's material.
The Properties panel lets you inspect and edit the properties of the material.
 - **Templates Panel**
You can add templates by selecting an object and add it.
 - **Properties Panel**
The property panel lets you inspect and change properties of currently selected Ogitor objects.
 - **Meshes Panel**
Shows all available meshes. Drag and drop a mesh into the main view to add it to the scene.
 - **Objects Panel**
All available object plugins are show on this panel. Drag and drop into the main view to add them to the scene.
- **Toggle buttons.**
These buttons lets you toggle the various tool panels on/off.
Also has a button to toggle full screen mode.
- **Statusbar.**
Contains a Camera menu which lets you retrieve or delete camera positions. Click the Save Position button in the Camera Options section of the main toolbar to store a camera position.
The statusbar also shows the current camera position and the frames per second rendered (FPS).



4. Controles

Texto de la documentación de Ogitor

Basic Movement

Basic Camera Movement: W: Forward, S: Back, A: Left, D: Right, Q: Up, E: Down

Rotate Camera: Right Mouse Button Click and move.

Rotate Camera Around Object*: ALT + Right Mouse Button Click and move.

Common Controls

Select Object: Left Mouse Button Click in Selection Mode or ALT + Left Mouse Button Click
(Always works as Select)

Set Focus to Object: ALT + Left Mouse Click on object or "F" key when there is already a selection.

Zoom in/out: Mouse Wheel.

Zoom in/out of Object: ALT + Mouse Wheel.

Delete Selected Object: DELETE key (Will ask for confirmation).

Changing View to Saved Camera Positions: CTRL + 0-9 for saved camera positions 0-9.

Mode Controls

Selecting Move / Rotate / Scale Axis : While using Move / Rotate / Scale tools, you can select the Axis to work on by selecting the appropriate Axis Pivot on the Gizmo.

END key changes transformation Axis during Move / Rotate / Scale operation.

Move

Locked translate axis: If no Axis is selected, default movement plane is the plane facing directly to the camera. Press END, or select the axis on the gizmo to translate the object on a fixed axis.

Cloning Objects: Hold SHIFT while dragging an object (in Render Window) to clone an object.

Snap to Grid: Hold down CONTROL to snap the object to the grid.

Rotate

Snap to Angle: Hold down CONTROL to snap the object to angle while rotating.

Scale


Uniform Scaling: Hold down CTRL key while scaling to scale on all Axes.

Terrain Smooth/Deform


SHIFT will reverse the operation when smoothing/deforming, changing from pushing to pulling vertices.

Otros

Modo de visualización

Podemos visualizar la escena en solid, wireframe o puntos usando el icono  de la barra de herramientas de render.

Capas (Layers)

Podemos visualizar o no visualizar las diferentes capas (layers) de la escena utilizando el icono  de la barra de herramientas de render.

Para el mundo de los sueños usaremos el layer 0 y para el mundo de las pesadillas el layer 1. Además habrá que adaptar los nombres de los objetos para definir cuál es el objeto del mundo de los sueños en el mundo de las pesadillas. Ver apartado Mundos paralelos y Game Objects.

5. Propiedades de la escena

Para acceder a las propiedades de la escena hay que seleccionar el objeto SceneManager en el menú de la izquierda en la pestaña scene.

Name: Nombre de la escena, en principio el mismo que el del nivel.

Position, Scale, Orientation y Config File no se usan.

Ambient: color de la luz ambiente de la escena

Propiedades del SkyBox

Active: indica si pinta o no.

Distance: Distancia del SkyBox respecto la escena.

Material: Material del SkyBox, para crear uno nuevo hay que hablar con programación.

Propiedades del SkyDome

Active: indica si pinta o no.

Material: Material del SkyDome, para crear uno nuevo hay que hablar con programación.

Propiedades de Fog

De momento nada.

Propiedades de Shadows

De momento nada.

6. Mundos paralelos

Layers

Para el mundo de los sueños usaremos el layer 0 y para el mundo de las pesadillas el layer 1.

Nombres de los objetos

Además habrá que adaptar los nombres de los objetos para definir cuál es el objeto del mundo de los sueños en el mundo de las pesadillas.

Cada objeto tendrá un nombre base (por ejemplo “tripollo” en el caso de los tripollos), un identificador y por último la letra “d” o “n” según se trate de dreams o nightmares. Cada campo tiene que ir separado por un asterisco y no se pueden repetir nombres. De modo que el formato sería:

nombre_objeto_d/n#id

Por ejemplo para el tripollo con identificador 15 deberíamos tener el objeto de sueños definido como

Tripollo_d#15 en el layer0

Y el tripollo de pesadillas como

Tripollo_n#15 en el layer1

Ambos objetos deben tener la misma posición para poder verlos correctamente en Ogitor. Para el juego se cogen los parámetros posicionales del mundo de los sueños (layer0) para ambos mundos.

7. Añadir un Modelo

Para añadir un modelo iremos a la pestaña **meses** del menú de la derecha, entonces arrastraremos el modelo que queramos a la escena. Si el modelo no está texturizado recomiendo añadir una luz a la escena para verlo correctamente (Ver sección 6)


Podemos modificar su posición rotación y escala con los controles del punto 4, apartado **Mode Controls**.

Podemos acceder a las propiedades del modelo que tenemos seleccionado en la pestaña **Properties** del menú de la derecha.

IMPORTANTE: Si añadimos un modelo de ese modo no tendrá ningún comportamiento específico. Ver apartado **Game Objects**

Propiedades de un modelo

Name: Nombre de la entidad de Ogre, ver apartado **Game Objects**.

Layer: Capa en la que se encuentra el modelo. Layer 0 para sueños y Layer 1 para pesadillas
Ver .

Position: Posición del modelo. Si el modelo se encuentra dentro de un nodo esta posición será relativa al nodo.

Orientation: Rotación del modelo expresada en un Quaternion de Ogre. Si el modelo se encuentra dentro de un nodo esta rotación será relativa al nodo.

Scale: Escala del modelo. Si el modelo se encuentra dentro de un nodo esta escala será relativa al nodo.

Tracking target: De momento no hace nada.

Mesh File: Fichero del modelo.

Cast shadows: Indica si el objeto hace sombras o no. Si lo activamos el objeto dará sombras en el juego, en Ogitor no se visualizan.

SubEntities: Lista de submodelos del modelo. En nuestro caso deberíamos tener un submodelo por cada textura que utilice el modelo. Si no es así la exportación no ha sido correcta y deberíamos hacer un Merge antes de exportar.

SubEntity#-Material: Material de Ogre (definido en el fichero .material) que utiliza el submodelo. No tiene porqué coincidir con el nombre del fichero. Más adelante en el proyecto definiremos que pautas seguimos para los nombres de los materiales.

SubEntity#-Visible: Para cada submodelo podemos indicar si es visible o no con el checkbox **Visible**.

8. Añadir una luz


Para añadir una luz debemos ir a la pestaña Objects del menú de la derecha y arrastrar el icono Light a la escena. **Para modificar la luz ambiente ver Propiedades de la Escena.**

Podemos modificar su posición rotación con los controles del punto 4, apartado Mode Controls. Escalar una luz no tiene ningún efecto.

Podemos acceder a las propiedades de la luz que tenemos seleccionado en la pestaña Properties del menú de la derecha.

Propiedades de una luz

Name: Nombre de Ogre, ver apartado Game Objects.

Layer: Capa en la que se encuentra. Layer 0 para sueños y Layer 1 para pesadillas. Ver .

Position: Posición. Si el modelo se encuentra dentro de un nodo esta posición será relativa al nodo.

Orientation: Rotación expresada en un Quaternion de Ogre. Si se encuentra dentro de un nodo esta rotación será relativa al nodo.

Direction: Dirección a la que emite esta luz. Solo válido para LT_SPOTLIGHT y LT_DIRECTIONAL

Light Type: Tipo de luz:

- LT_POINT: (Punto de luz) emite en todas direcciones desde el punto donde se encuentra la luz.
- LT_DIRECTIONAL: (Direccional) emite luz en la dirección especificada en toda la escena, no tiene en cuenta la posición de la luz.
- LT_SPOTLIGHT: (Tipo linterna) emite luz en la dirección especificada y solo en el rango definido en Range. NOTA: En Ogitor no se pueden visualizar los resultados para este tipo de luz, hay que verlas dentro del juego.

Diffuse: Color difuso de la luz, para modificarlo hay que seleccionar el campo 'Diffuse' (no al de los colores), entonces aparece un botón con ..., lo pulsamos y seleccionamos el color de la luz.



Specular: Color especular de la luz, de modifica del mismo modo que el diffuse.

Range: Solo para luces LT_SPOTLIGHT. Os pongo lo que pone en la documentación de Ogre
the angle of the inner and outer cones and the rate of falloff between them.

innerAngle [Angle](#) covered by the bright inner cone The inner cone applicable only to Direct3D, it'll always treat as zero in OpenGL.

outerAngle [Angle](#) covered by the outer cone

falloff The rate of falloff between the inner and outer cones. 1.0 means a linear falloff, less means slower falloff, higher means faster falloff.

NOTA: En Ogitor no se pueden visualizar los resultados para este tipo de luz, hay que verlas dentro del juego.

Attenuation:

NOTA: En Ogitor no se pueden visualizar los resultados de la atenuación, hay que verlas dentro del juego.

Os pongo lo que pone en la documentación de Ogre: Sets the attenuation parameters of the light source i.e. how it diminishes with distance.

Remarks:

Lights normally get fainter the further they are away. Also, each light is given a maximum range beyond which it cannot affect any objects.

Light attenuation is not applicable to directional lights since they have an infinite range and constant intensity.

This follows a standard attenuation approach - see any good 3D text for the details of what they mean since i don't have room here!

Parameters:

- range* The absolute upper range of the light in world units (distancia a la que llega la luz)
- constant* The constant factor in the attenuation formula: 0.0 means never attenuate, 1.0 is complete attenuation
- linear* The linear factor in the attenuation formula: 1 means attenuate evenly over the distance
- quadratic* The quadratic factor in the attenuation formula: adds a curvature to the attenuation formula.

Para entender cómo funciona poned el Range a 25, por ejemplo y el resto todo a 0. Esto quitara toda la atenuación de la luz. Entonces id añadiendo atenuacion poco a poco (de 0.1 en 0.1) con los distintos valores. Primero probad con todos los valores de constant luego linear y luego quadratic.



En todas las imágenes la luz es del mismo color diffuse (1,0,0) specular (0,0,0). De izquierda a derecha, en la primera los parámetros son (25,0,0,0) en la segunda (25,0.7,0,0) y en la tercera (25,0.7,0.1,0).

Power: Indica la potencia de una luz respecto al resto. Por ejemplo una luz con power 0.5 da 4 veces menos luz que una con power 2.

NOTA: Los efectos de este parámetro en Ogitor no son los mismos que en el juego, hay que ver el juego en funcionamiento para ver los resultados.

No sé si lo vamos a usar, de momento lo dejaremos a 1 y usaremos la attenuation.

Cast Shadows: Indica si la luz emite sombras. Si lo activamos la luz dará sombras en el juego, en Ogitor no se visualizan.

9. Añadir un sistema de partículas

Para añadir un sistema de partículas debemos ir a la pestaña Objects del menú de la derecha y arrastrar el icono Particle a la escena. Solo podremos poner el sistema de partículas en la escena, no modificarlo. Para eso debemos utilizar el editor de sistema de partículas.

Podemos modificar su posición rotación y escala con los controles del punto 4, apartado Mode Controls.

Podemos acceder a las propiedades del que tenemos seleccionado en la pestaña Properties del menú de la derecha.

IMPORTANTE: Si añadimos sistema de partículas de ese modo no tendrá ningún comportamiento específico. Ver apartado Game Objects

Propiedades de un sistema de partículas

Name: Nombre de Ogre, ver apartado Game Objects.

Layer: Capa en la que se encuentra. Layer 0 para sueños y Layer 1 para pesadillas. Ver .

Position: Posición del modelo. Si se encuentra dentro de un nodo esta posición será relativa al nodo.

Orientation: Rotación expresada en un Quaternion de Ogre. Si se encuentra dentro de un nodo esta rotación será relativa al nodo.

Scale: Escala. Si se encuentra dentro de un nodo esta escala será relativa al nodo.

Particle System: Tipo del sistema de partículas (se cargan de los ficheros .particle). Para editarlos hay que hacerlo en el editor de sistemas de partículas.

Tracking target: De momento no hace nada.

Cast shadows: Indica si el objeto hace sombras o no. Si lo activamos el objeto dará sombras en el juego, en Ogitor no se visualizan.

10. Añadir un billboard set

Para añadir un billboard set debemos ir a la pestaña Objects del menú de la derecha y arrastrar el icono Billboard Set a la escena.

Un billboard set consiste en una serie de planos (billboards) que giran alrededor de un punto de modo que siempre están mirando a la cámara. Para añadir un billboard al billboard set usad el botón derecho encima de "Billboards". Por defecto los billboard sets tienen un billboard "billboard0".


Podemos modificar su posición rotación y escala con los controles del punto 4, apartado Mode Controls.

Podemos acceder a las propiedades del que tenemos seleccionado en la pestaña Properties del menú de la derecha.

IMPORTANTE: Si añadimos un billboard set de ese modo no tendrá ningún comportamiento específico. Ver apartado Game Objects

Propiedades comunes para todos los billboards del billboard set

Name: Nombre de Ogre, ver apartado Game Objects.

Layer: Capa en la que se encuentra. Layer 0 para sueños y Layer 1 para pesadillas. Ver .

Position: Posición. Si se encuentra dentro de un nodo esta posición será relativa al nodo.

Orientation: Rotación expresada en un Quaternion de Ogre. Si se encuentra dentro de un nodo esta rotación será relativa al nodo.

Scale: Escala. Si se encuentra dentro de un nodo esta escala será relativa al nodo.

Tracking target: De momento no hace nada.

Billboard Type: Indica como el billboard se orienta a la cámara a partir billboard Origin, probad los diferentes modos.

Billboard Origin: Indica el punto desde dónde el billboard se orienta a la cámara, probad los diferentes modos.

Billboard Rotation: Permite elegir si son los vértices o las coordenadas de textura las que definen la orientación del billboard, probad los diferentes modos.

Default Width/Height: Dimensiones por defecto de los billboards que tiene el billboard set.

Material: Material del billboard set, que definirá su textura. Para crear un material de momento hay que hablar con algún programador :P

Point Rendering: Tiene que estar desactivado.

Render Distance: Distancia máxima respecto a la cámara a la que se pintará el billboard. A partir de esa distancia no se pintará.

Cast shadows: Indica si el objeto hace sombras o no. Si lo activamos el objeto dará sombras en el juego, en Ogitor no se visualizan.

Propiedades específicas para cada billboard

Un billboard set puede tener uno o más billboards. Para añadir un billboard al billboard set usad el botón derecho encima de "Billboards". Estos son los parámetros específicos para cada uno de estos billboards.

Position: Posición relativa a la posición del billboard set.

Dimensions: Dimensiones del billboard. Cuando se crea un billboard si no especificamos nada serán "Default Width/Height".

Colour: Color del billboard, no tiene mucho sentido debido a que el material lo sobrescribe...

Rotation: Rotación relativa a la rotación del billboard set.

TexCoordIndex: Cuando lo modificas Ogitor explota y se pierden los cambios =_= ¡ No tocar !

TexRect: Coordenadas de Textura del billboard. U1=0 V1=0 U2=1 V2=1 debería servir.

11. Añadir una cámara

Utilizaremos las cámaras para definir cámaras fijas, por ejemplo si en una zona queremos utilizar una cámara fija en vez de la de tercera persona habrá que definir una cámara en esa posición.


Para añadir una cámara debemos ir a la pestaña Objects del menú de la derecha y arrastrar el icono Camera a la escena.

Podemos modificar su posición rotación y escala con los controles del punto 4, apartado Mode Controls.

Podemos acceder a las propiedades del que tenemos seleccionado en la pestaña Properties del menú de la derecha.

Propiedades comunes para todos los billboards del billboard set

Name: Nombre de Ogre, ver apartado Game Objects.

Layer: Capa en la que se encuentra. Layer 0 para sueños y Layer 1 para pesadillas. Ver .

View Mode: Hay que dejarlo en FREE.

Position: Posición. Si se encuentra dentro de un nodo esta posición será relativa al nodo.

Orientation: Rotación expresada en un Quaternion de Ogre. Si se encuentra dentro de un nodo esta rotación será relativa al nodo.

Tracking target: De momento no hace nada.

Auto aspect ratio: Hay que desactivarlo, ¡por defecto viene activado!

Clip distance: distancia mínima (near) y máxima (far) del frustrum de visión. Lo que no esté entre las dos distancias respecto a la cámara no se pintará.

FOV: (o FOVy) campo de visión. Ogitor no usa grados para definirlo. Por defecto es 1, lo que equivale a 55º, puede ir de 0º (0) a 180º (3.73). A más de 3.73 será 180º y a menos de 0, 0º.

PolygonMode: Modo de visualización de la escena, en principio hay que dejarlo en PM_SOLID aunque para alguna cámara de debug podemos usar los otros modos.

12. Game Objects

Hasta ahora los objetos que hemos visto no tienen ningún comportamiento en el juego, simplemente se pintan y ya está. Para asignar un comportamiento será necesario que tengan el nombre del objeto del juego correcto.

Nota: En principio para modelar el escenario solo es necesario tener en cuenta el nombre del objeto, los ficheros adicionales que se definen en la lista son información para los programadores.

Sobre los nombres de los objetos

El comportamiento de un Game Object vendrá dado por su nombre. Ver cada game object para saber que nombre es necesario que tenga.

Recordad que hay que seguir el formato nombre_objeto_d/n #id definido en el apartado 6.

Listado de nombres de los GameObjects y su comportamiento

Nota: Los ficheros .mesh y .material de los modelos deberían tener el mismo nombre que el de los Game Objects para que se ponga el nombre automáticamente.

- **ony:** Ogitor Entity. en principio solo debería estar definido en sueños (si no hay cambios de modelo de un mundo a otro). También indica la posición desde dónde se empieza el nivel.
- **entity:** Ogitor Entity. Cualquier objeto que sea un modelo no necesite ningún comportamiento y se pueda traspasar andando debería tener ese nombre
- **tripollo:** Ogitor Entity.
- **terrain:** Ogitor Entity el fichero nxs debe tener el mismo nombre que el fichero .mesh del terrain. Todos los terrenos y objetos por los cuales ony puede andar por encima deberían tener ese tipo
- **light:** Ogitor Light, es una luz inmóvil, recordad que se ve distinto en el juego que en el editor.
- **camera:** Ogitor camera
- **billboard-set:** Ogitor BillboardSet, billboard de cualquier tipo
- **particle-system:** Ogitor ParticleSystem, sistema de partículas
- **portal:** Ogitor Entity objeto de cambio de mundos
- **eye:** Ogitor Entity enemigo ojo
- **item-1up:** Ogitor Entity
- **item-maxhp**
- **volume-box:** No se pinta por pantalla, se trata de una zona de evento
- **volume-capsule:** No se pinta por pantalla, se trata de una zona de evento

De momento hay estos, si ponéis otros objetos que tienen un comportamiento (como otros enemigos etc...) ponedles un nombre en esa línea.