Tutorial de Física

Sumario

Ver los anteriores tutoriales básicos para más información sobre la creación básica de objetos, manejo del reloj, jerarquía de fotogramas, animaciones, cámaras & vistas, música & sonido y efectos(FXs).

Este tutorial muestra como añadirle propiedades físicas a los objetos y manejar colisiones.

Como puedes ver, las propiedades físicas son completamente manipuladas por datos. Así, creando un objeto con propiedades físicas (ej. con un cuerpo) o sin ellas, resulta en exactamente la misma linea de código.

Los objetos pueden ser enlazados a un cuerpo, que puede ser estático o dinámico.
Cada cuerpo puede estar hecho de hasta 8 partes.

Un cuerpo es definido por:

- su forma (actualmente caja, esfera y malla(ej. poligono convexo) son los únicos disponibles)
- información acerca del tamaño de la forma (esquinas para la caja, centro y radio para la esdera,)

A body part is defined by:

- its shape (currently box, sphere and mesh (ie. convex polygon) are the only available)
- information about the shape size (corners for the box, center and radius for the sphere, vertices for the mesh)
- if no size data is specified, the shape will try to fill the complete body (using the object size and scale)
- collision “self” flags that defines this part
- collision “check” mask that defines with which other parts this one will collide [1)]
- a flag (`Solid`) specifying if this shaped should only give information about collisions or if it should impact on the body physics simulation (bouncing, etc...)
- various attributes such as restitution, friction, density, ...

In this tutorial we create static solid walls around our screen. We then spawn boxes in the middle.
The number of boxes created is tweakable through the config file and is 100 by default.

The only interaction possible is using left and right mouse buttons (or left and right keys) to rotate the camera.
As we rotate it, we also update the gravity vector of our simulation.
Doing so, it gives the impression that the boxes will be always falling toward the bottom of our screen no matter how the camera is rotated.

We also register to the physics events to add a visual FXs on two colliding objects.
By default the FX is a fast color flash and is, as usual, tweakable in realtime (ie. reloading the config history will apply the new settings immediately as the FX isn't kept in cache by default).

Updating an object scale (including changing its scale with FXs) will update its physical properties (ie.

its body).
Keep in mind that scaling an object with a physical body is more expensive as we have to delete the current shapes and recreate them at the correct size.
This is done this way as our current single physics plugin is based on Box2D which doesn't allow realtime rescaling of shapes.

This tutorial does only show basic physics and collision control, but, for example, you can also be notified with events for object separating or keeping contact.

Detalles

Recursos

[1)]

two parts in the same body will never collide

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
https://www.orx-project.org/wiki/es/orx/tutorials/physics?rev=1330789397

Last update: **2025/09/30 17:26 (14 months ago)**