



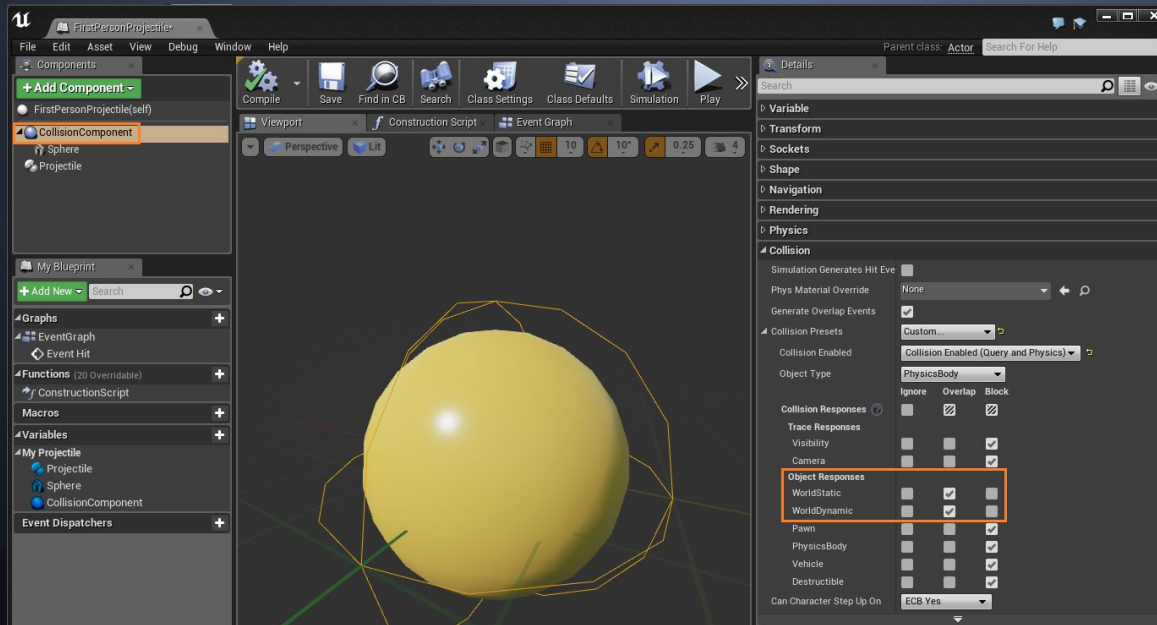
Desarrollo de Videojuegos

Contenido del nivel y
programación de la dinámica
Simulación física

Motivación

- ¿Cómo sé si *el avatar del jugador ha cogido un potenciador*, o si *debe caerse porque no apoya en el suelo*, o si *lo estoy señalando con el ratón*?

...



Motivación

- Para un juego es casi tan importante *simular lo físico* como *lo visible*
- Para simular **cuerpos rígidos**, Unreal usa su propio sistema: **Chaos Physics**
 - Interacción física entre varios objetos, destrucción masiva dinámica, tejidos, cabellos, fluidos, vehículos...
- La **detección de colisiones** con precisión va aparte



Colisión

- El concepto de **colisión** es de los aspectos más importantes a conocer de Unreal Engine, junto a sus posibles **respuestas** y las **configuraciones preestablecidas** existentes



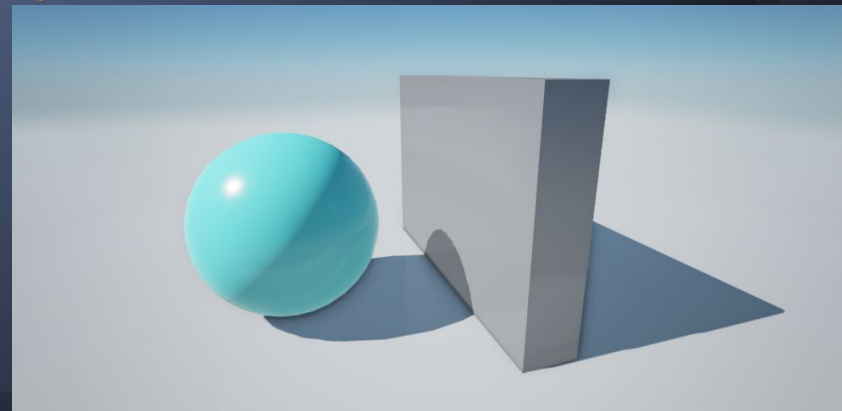
Colisiones y proyecciones

Collision Overview

An overview of how Collision and Collision Responses operate in Unreal Engine 4.

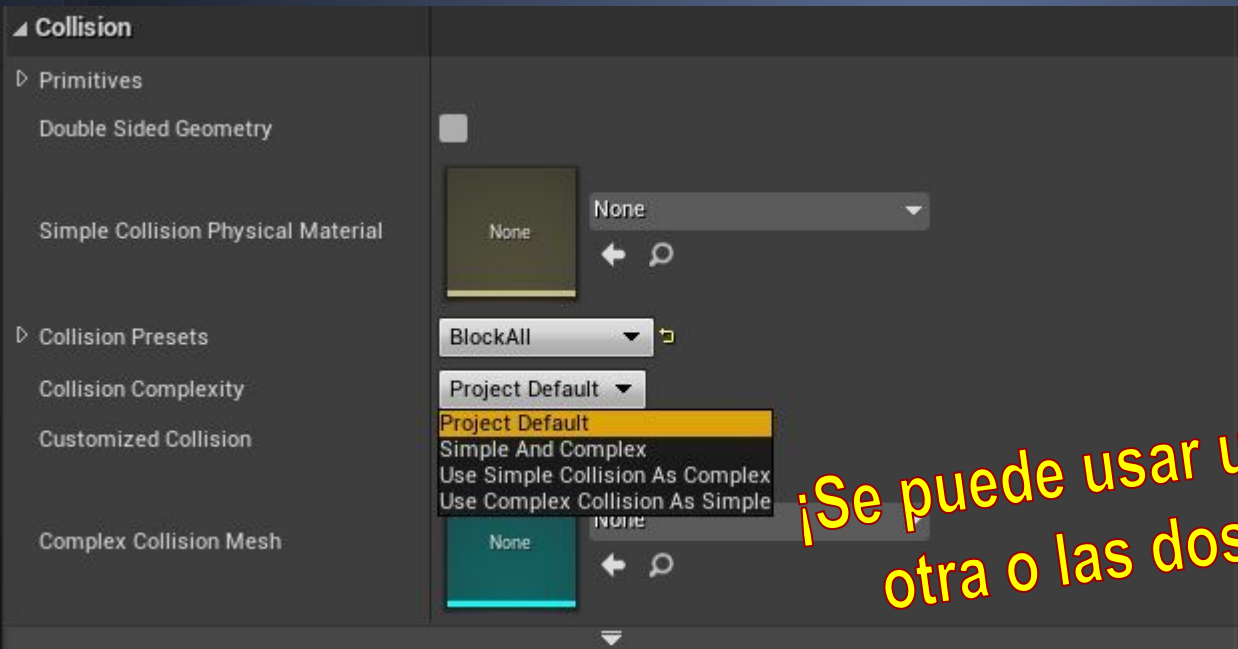
Unreal Engine 4.22

<https://docs.unrealengine.com/en-US/Engine/Physics/Collision/Overview/index.html>



Formas de colisión

- Toda malla tiene **colisión compleja** (¡tris de su propia geometría!) y *puede* tener una **colisión simple** (es fácil hasta **autogenerarla**)
 - Las **consultas de colisión** funcionan con ambas formas, pero la simulación de cuerpos rígidos sólo con la simple



¡Se puede usar una,
otra o las dos!

Respuestas a la colisión

COLLISION RESPONSES

- Un actor que colisiona tiene **volumen de colisión** y **“tipo” de objeto de colisión**
- Hay varias **respuestas** según el **“tipo” de objeto de colisión** del otro actor con que colisiona
 - **Bloquear** a ese otro actor
 - **Chocar** incluso, si alguno está en movimiento
 - **Solaparse** con ese otro actor
 - **Ignorarlo** por completo (visualmente parecen *solaparse*, pero no se genera ningún evento)



Puedes añadir los tuyos...

Respuestas a la proyección

TRACE RESPONSES

- La **proyección de rayos** es similar: *cada traza* tiene su **canal de traza de colisión**
- Cada actor define su **respuesta** según el *canal de traza de colisión* de la traza proyectada sobre él
 - Ej. **Visibilidad y cámara**
 - Técnica habitual para *predecir colisiones* y otros usos...

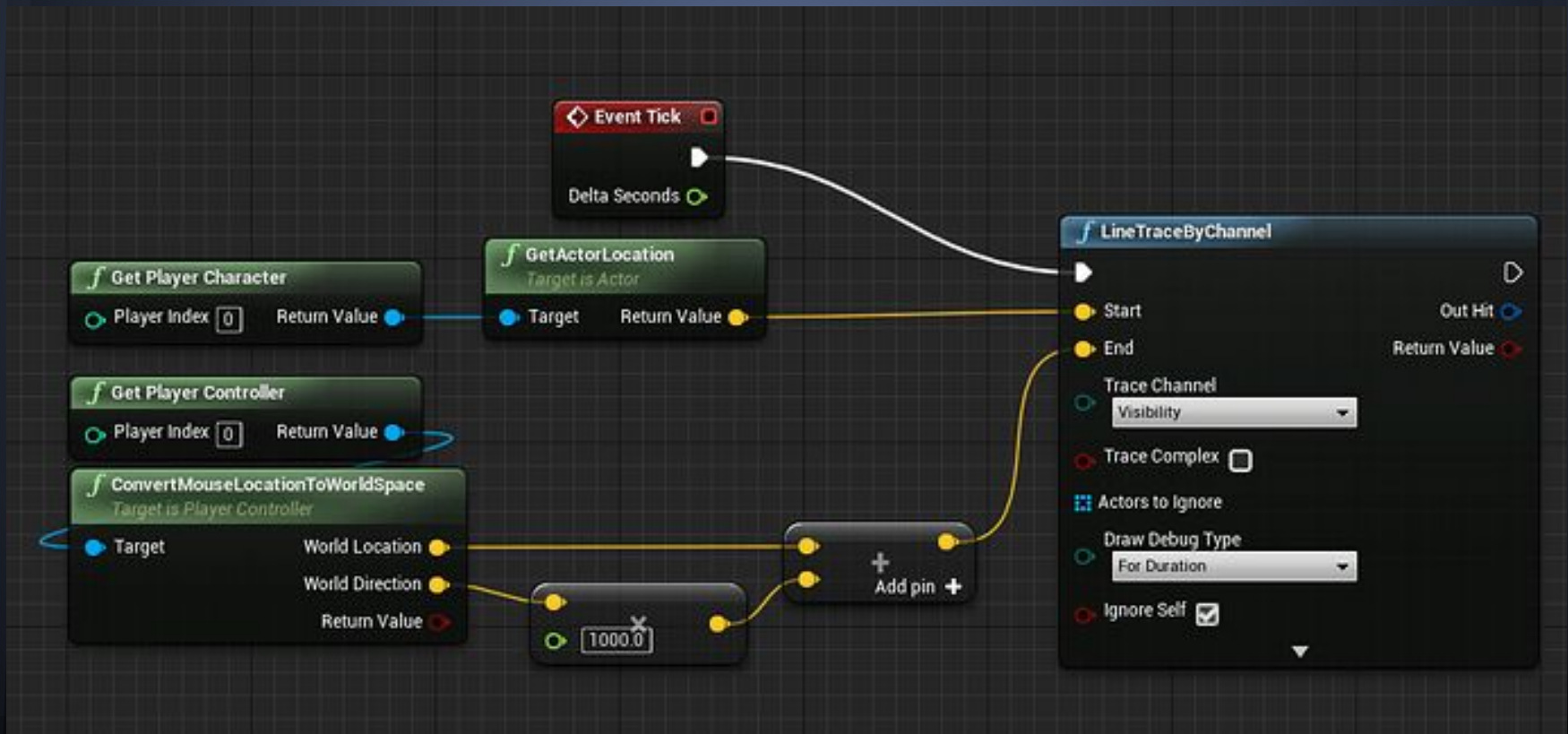


Puedes añadir los tuyos...

<https://docs.unrealengine.com/en-US/Engine/Physics/Tracing/index.html>

Ejemplo

- Proyectar una línea desde el pivote de mi personaje 10 metros en dirección a donde mira mi cámara



Interacciones comunes

Collision Overview

An overview of how Collision and Collision Responses operate in Unreal Engine 4.

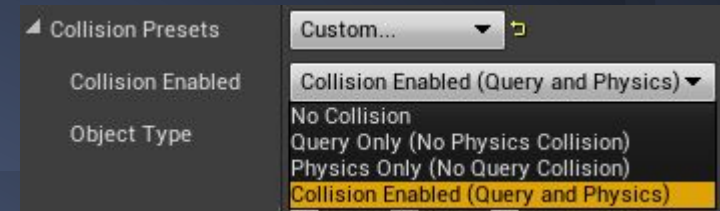
Unreal Engine 4.22

Collision Responses and **Trace Responses** form the basis for how Unreal Engine 4 handles collision and ray casting during run time. Every object that can collide gets an **Object Type** and a series of responses that define how it interacts with all other object types. When a collision or overlap event occurs, both (or all) objects involved can be set to affect or be affected by blocking, overlapping, or ignoring each other.

Trace Responses work basically the same way, except the trace (ray cast) itself can be defined as one of the Trace Response types, thusly allowing Actors to either block it or ignore it based on *their* Trace Responses.

<https://docs.unrealengine.com/en-US/Engine/Physics/Collision/Overview/index.html>

Interacciones comunes



- Sólo hay colisión si se **activan las colisiones**
 - Ya sea para **consultas geométricas** (trazas, barridos, solapamientos..), para **simulación física** (gravedad, rebotes, muñecas de trapo...) o ambas
- Sólo hay bloqueo si los dos actores **se bloquean entre sí**
 - Para recibir (muchos) eventos simulando física, activa **Simulation Generates Hit Events** en el actor
- Sólo hay solapamiento si los dos **se solapan entre sí** (o uno bloquea y otro solapa)
 - ¡Activa **Generate Overlap Events** en el actor donde quieras recibir el evento!

Interacciones comunes

Collision

Notify Rigid Body Collision ☐

Always Create Physics State ☐

Generate Overlap Events ☐

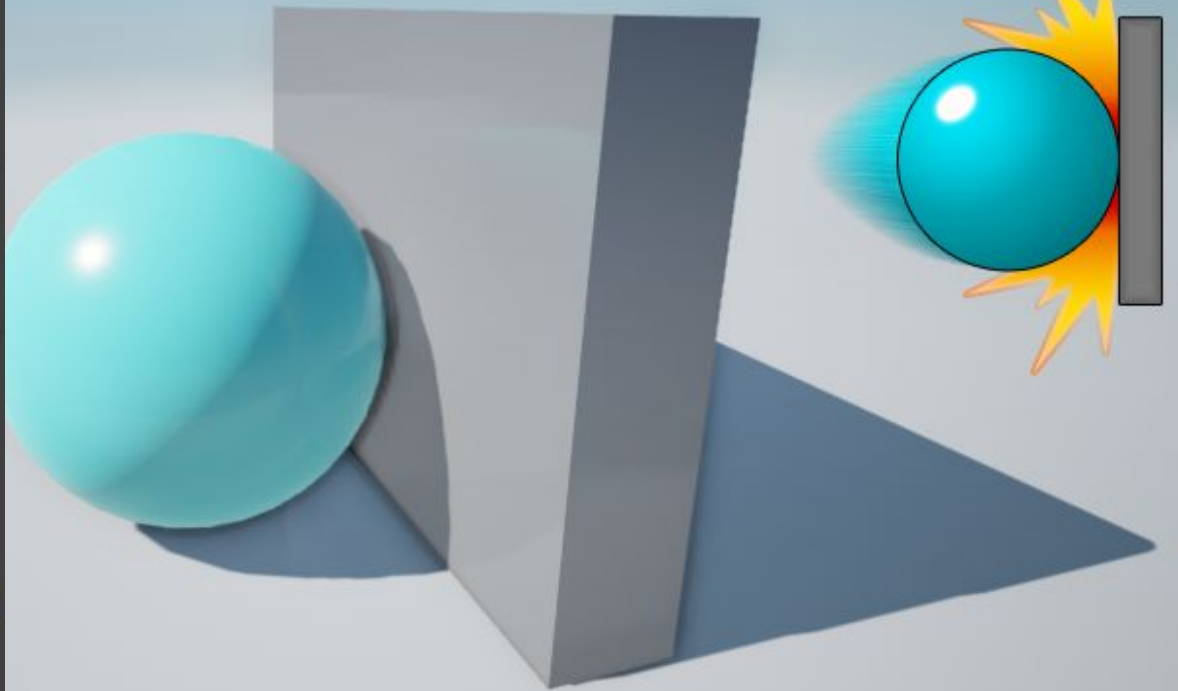
Trace Complex on Move ☐

Collision Presets **None**

Collision Enabled **Collision Enabled**

Object Type **PhysicsBody**

	Ignore	Overlap	Block
Collision Responses	☐	☒	☒
Trace Responses			
Visibility	☐	☒	☐
Camera	☐	☒	☐
Object Responses			
WorldStatic	☐	☐	☒
WorldDynamic	☐	☐	☒
Pawn	☐	☒	☐
PhysicsBody	☐	☒	☐
Vehicle	☐	☒	☐
Destructible	☐	☒	☐

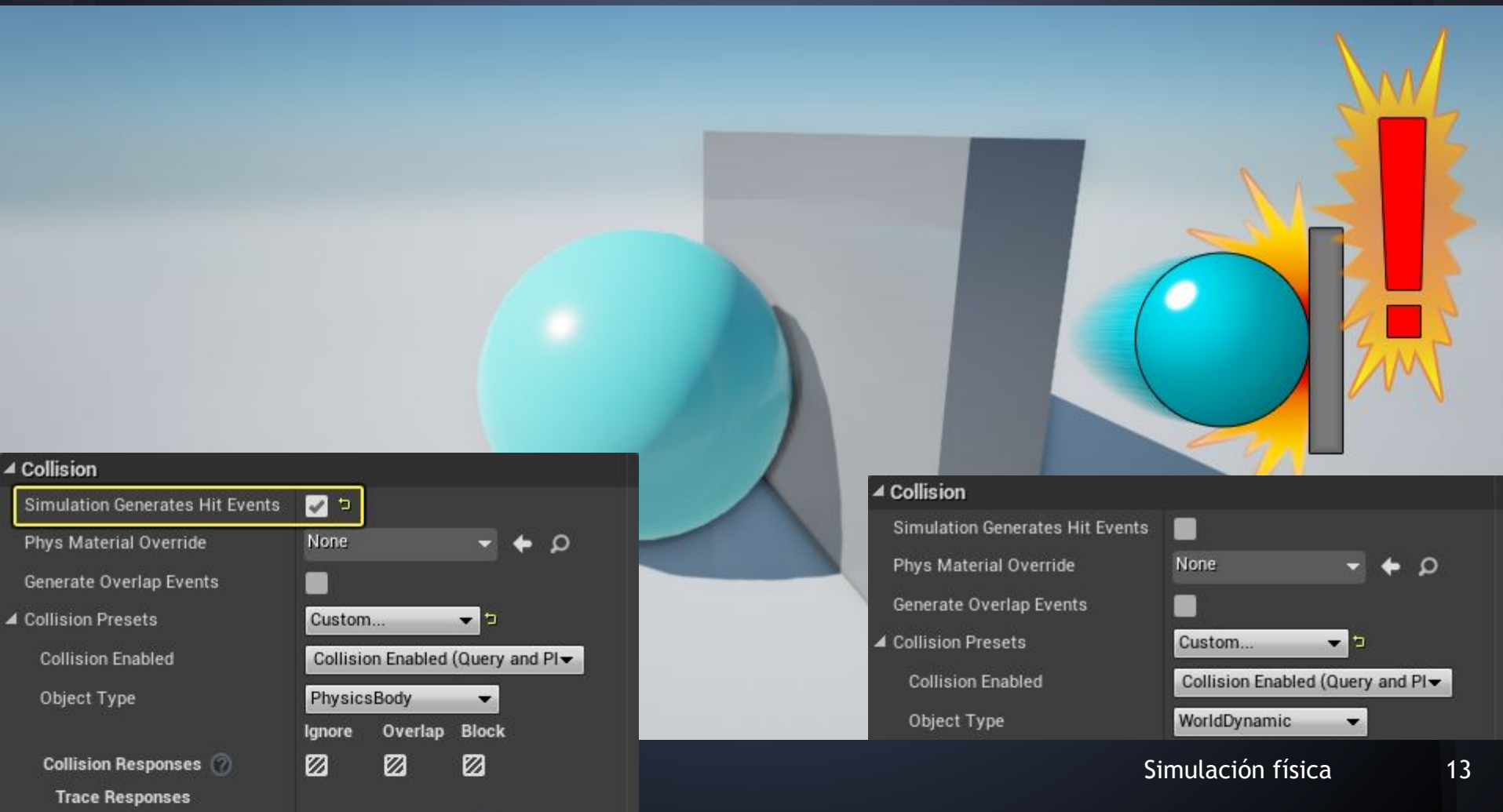


Object Type **WorldDynamic**

	Ignore	Overlap	Block
Collision Responses	☐	☒	☒
Trace Responses			
Visibility	☐	☐	☒
Camera	☐	☐	☒
Object Responses			
WorldStatic	☐	☐	☒
WorldDynamic	☐	☒	☐
Pawn	☐	☒	☐
PhysicsBody	☐	☐	☒
Vehicle	☐	☒	☐
Destructible	☐	☒	☐

Can Character Step Up On **ECB Yes**

Interacciones comunes



Interacciones comunes

Collision Responses ?

Trace Responses

Visibility

Camera

Object Responses

WorldStatic

WorldDynamic

Pawn

PhysicsBody

Vehicle

Destructible

	Ignore	Overlap	Block
	☐	☒	☒
Trace Responses			
Visibility	☐	☐	☒
Camera	☐	☐	☒
Object Responses			
WorldStatic	☐	☐	☒
WorldDynamic	☐	☒	☐
Pawn	☐	☒	☐
PhysicsBody	☐	☐	☒
Vehicle	☐	☒	☐
Destructible	☐	☒	☐

Collision

Notify Rigid Body Collision



Always Create Physics State



Generate Overlap Events



Trace Complex on Move



Collision Presets

None

Collision Enabled

Collision Enabled

Object Type

PhysicsBody

Collision Responses ?

Trace Responses

Visibility

Camera

Object Responses

WorldStatic

WorldDynamic

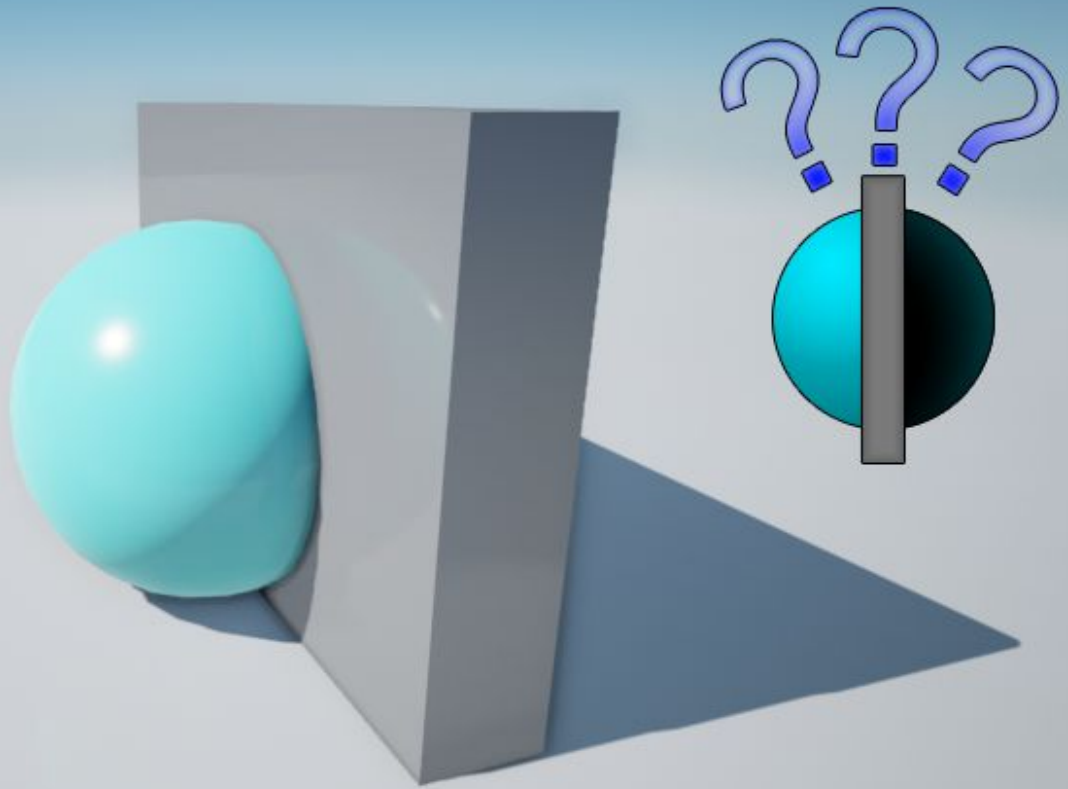
Pawn

PhysicsBody

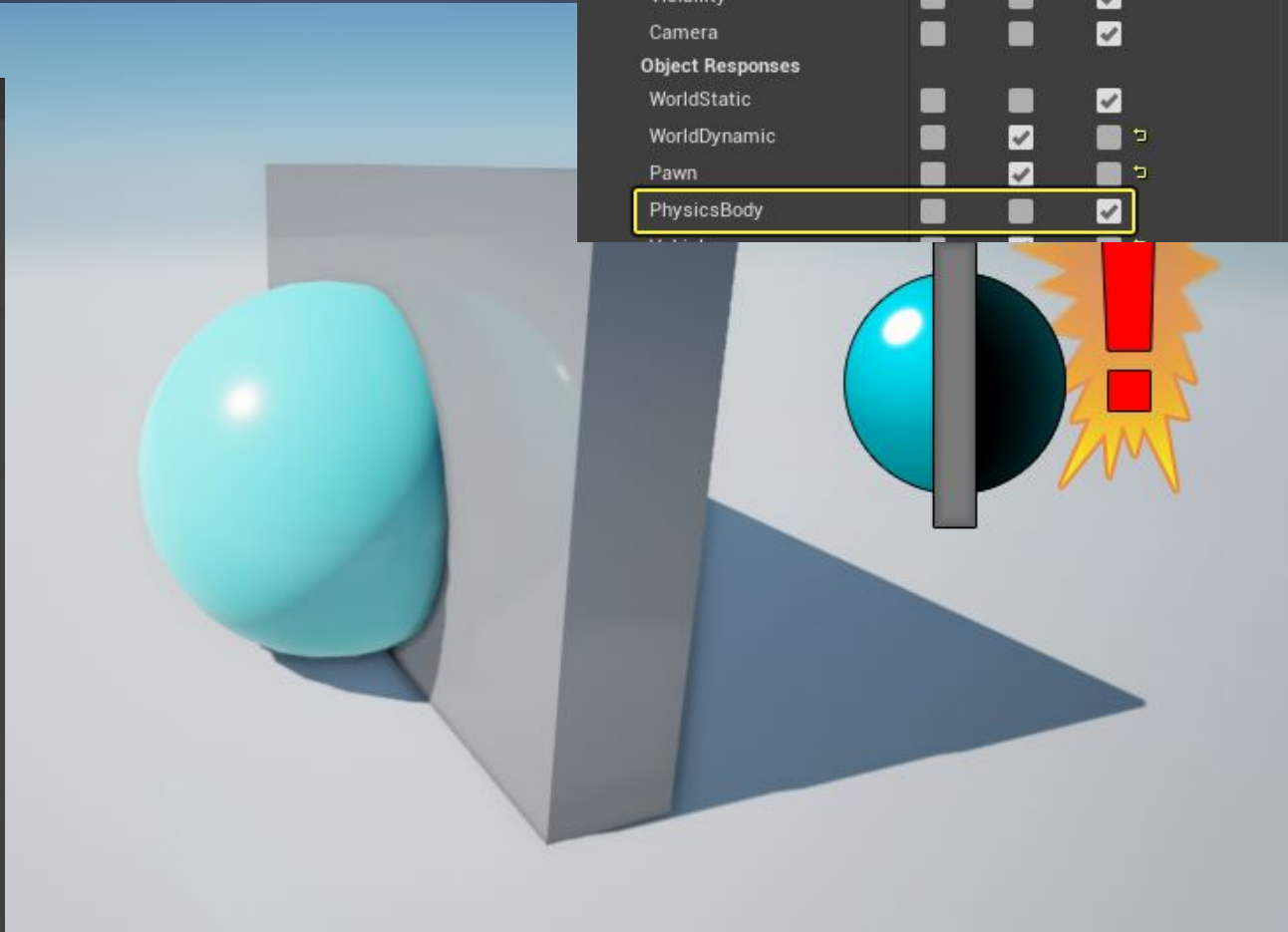
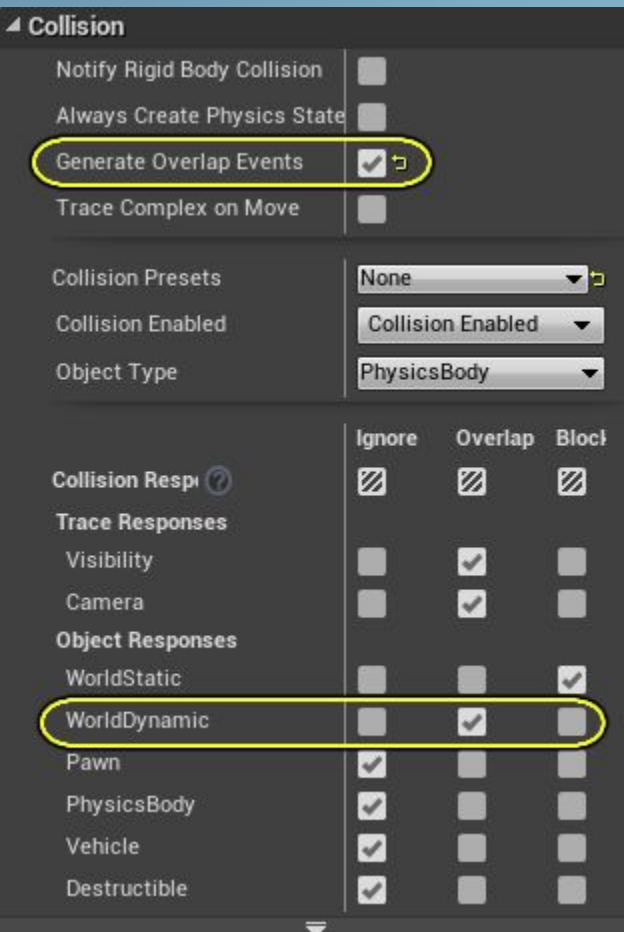
Vehicle

Destructible

	Ignore	Overlap	Block
	☒	☒	☒
Trace Responses			
Visibility	☐	☒	☐
Camera	☐	☒	☐
Object Responses			
WorldStatic	☐	☐	☒
WorldDynamic	☐	☒	☐
Pawn	☒	☐	☐
PhysicsBody	☒	☐	☐
Vehicle	☒	☐	☐
Destructible	☒	☐	☐



Interacciones comunes



Configuraciones preestablecidas

- En la web encontramos estas interacciones comunes, cómo crear formas de colisión y configuraciones preestablecidas

Collision Response Reference

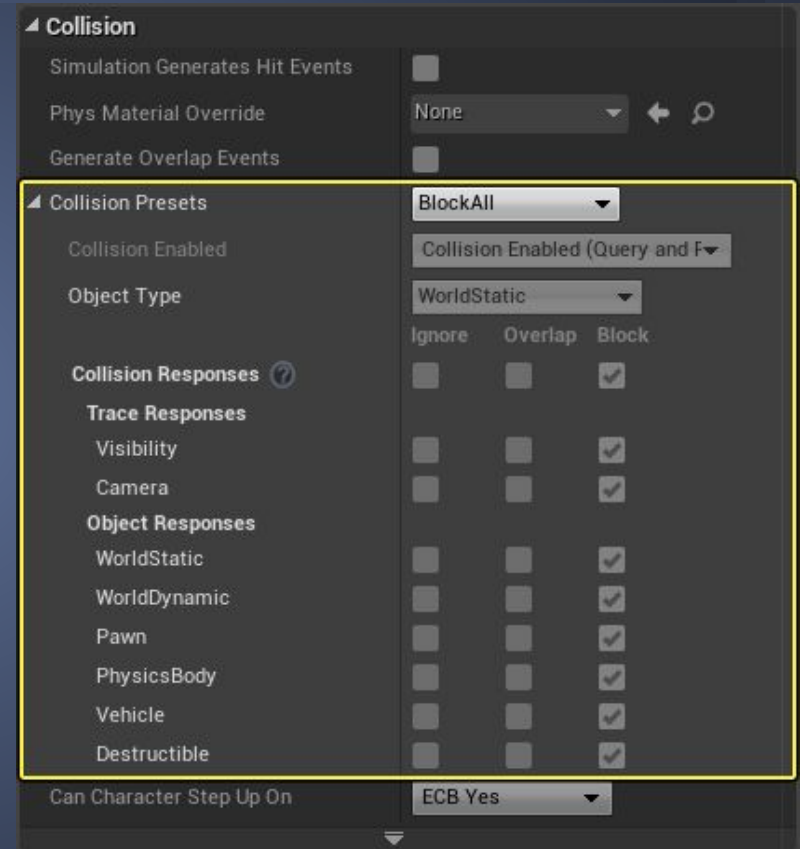
The Collision Presets section in the Collision Property Category has a large number of properties broken out here for clarity.

Unreal Engine 4.9

<https://docs.unrealengine.com/en-US/Engine/Physics/Collision/HowTo/index.html>

Configuraciones preestablecidas

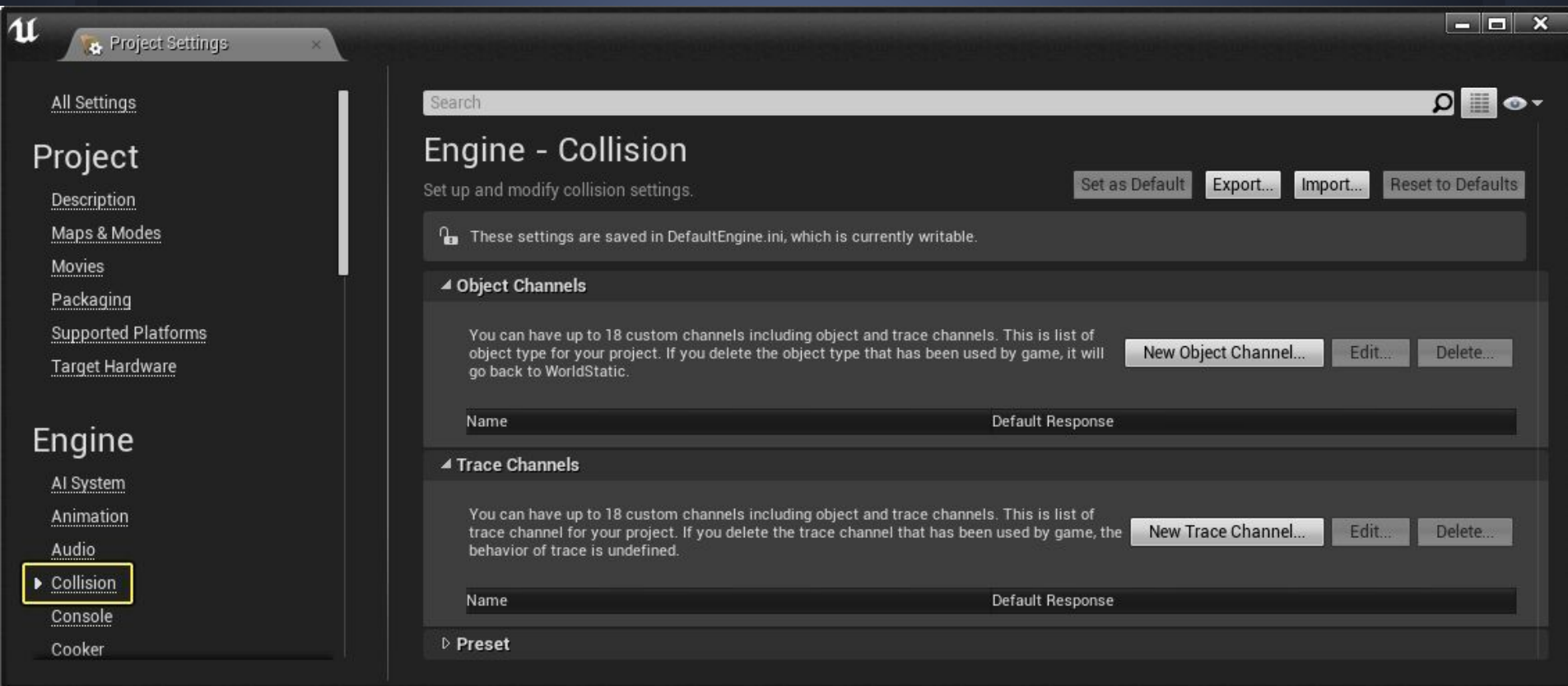
- Todas las propiedades que podemos elegir, también para crear **configuraciones *ad hoc***
 - Indicando una a una las posibles interacciones con objetos de todo tipo



<https://docs.unrealengine.com/en-US/Engine/Physics/Collision/Reference/index.html>

Canales de colisión

- Hasta 18 canales, sean de **objeto** o de **traza**



Simulación física

- Toda la física la *simula Unreal*, pero tú puedes definir **cueros físicos**, establecer **restricciones a los movimientos**, etc.

PhysX Constraint User Guide

Physical joints used to connect two rigid body simulated objects together.

Unreal Engine 4.9

Beginner

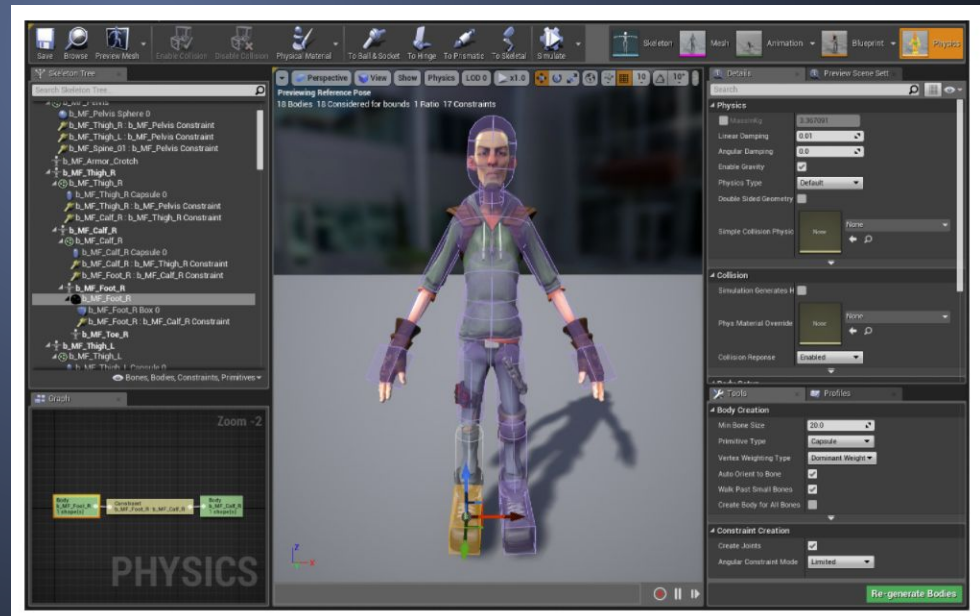


Unreal nos oculta los "micropasos" físicos, de frecuencia constante

<https://docs.unrealengine.com/en-US/Engine/Physics/Constraints/ConstraintsBlueprints/index.html>

Cuerpos físicos

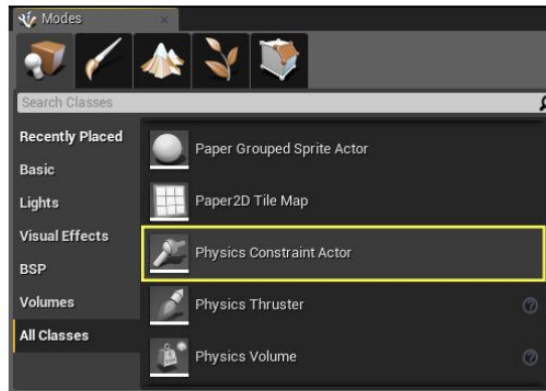
- Si un actor es **móvil** y tiene **colisión simple**... se puede activar **Simulate Physics** y se comportará como cuerpo físico
 - Para simular física en asuntos más complejos, como las **mallas esqueléticas**, existe un editor
 - **Physics Asset Tool (PhAT)**



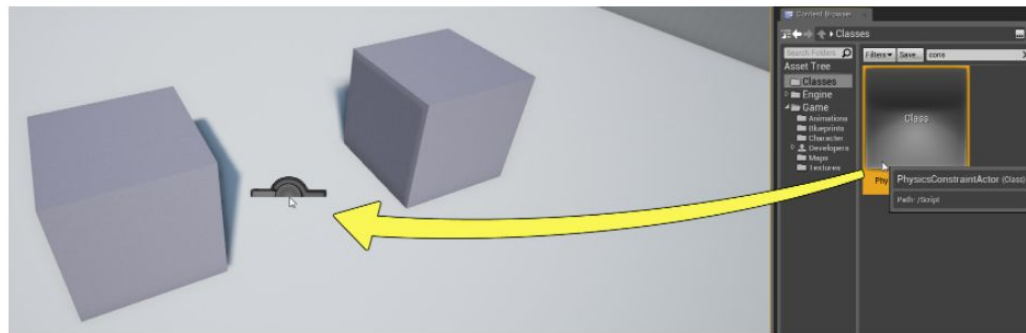
Restricciones físicas

Physics Constraint Actor

1. The Physics Constraint Actor can be found in the **All Classes** tab of the **Place Actors** panel:



2. From here, it can be brought into the level, like any other Actor, by clicking and dragging it into the viewport.



<https://docs.unrealengine.com/en-US/Engine/Physics/Constraints/index.html>

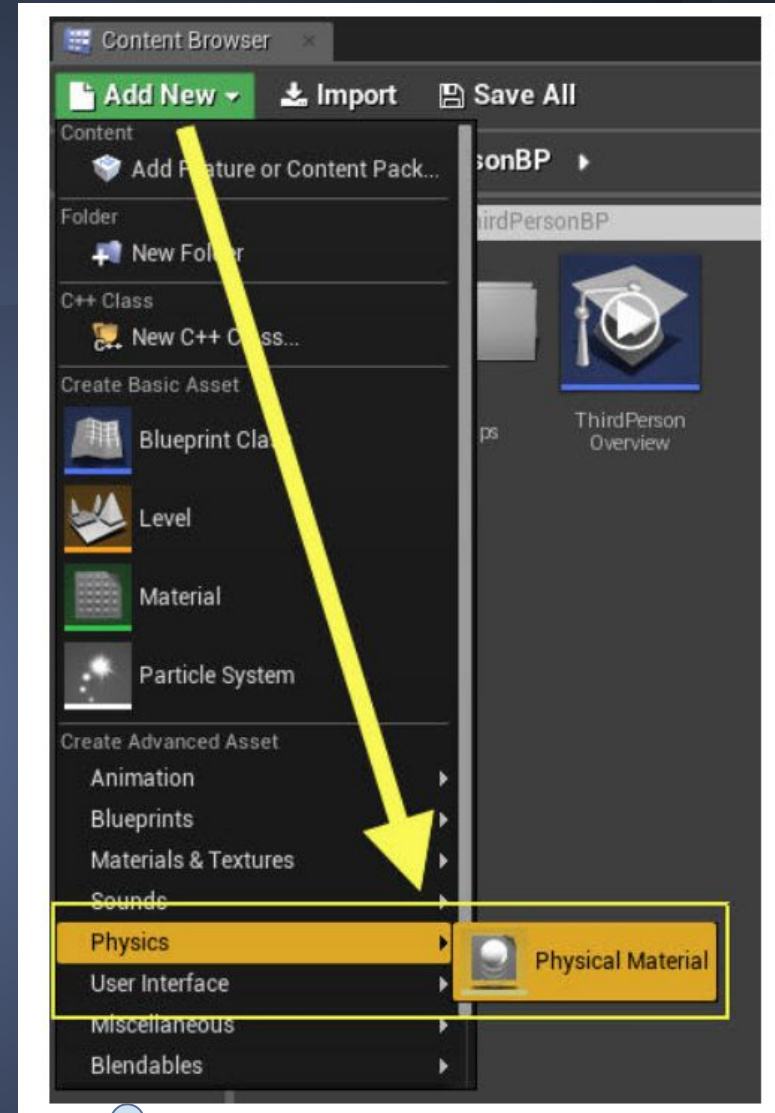
Participación

- ¿Para qué sirve el atributo **Generate Overlap Events** en la categoría de colisión?
 - A. Para que el actor salga solapado con otros actores
 - B. Para que el actor *pueda* solaparse con otros actores
 - C. Para que al solapar dos actores ambos reciban un evento
 - D. Para que si se solapa, el motor llame a su evento Overlap
 - E. Para que si se solapa, el otro actor reciba un evento



Materiales físicos

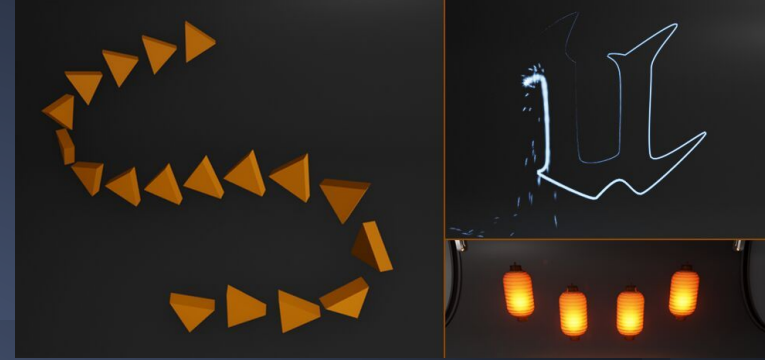
- Son recursos que se crean indicando **fricción estática y dinámica**, **restitución** (rebote), **densidad**, **sonidos**, **resistencia al impacto** o la **destrucción...**
- Estos se pueden añadir a **materiales** normales (y actuar así sobre *superficies*) o directamente al **colisionador** de un actor que **simula física**



Se pueden interpolar

<https://docs.unrealengine.com/en-US/Engine/Physics/PhysicalMaterials/index.html>

Curvas Spline

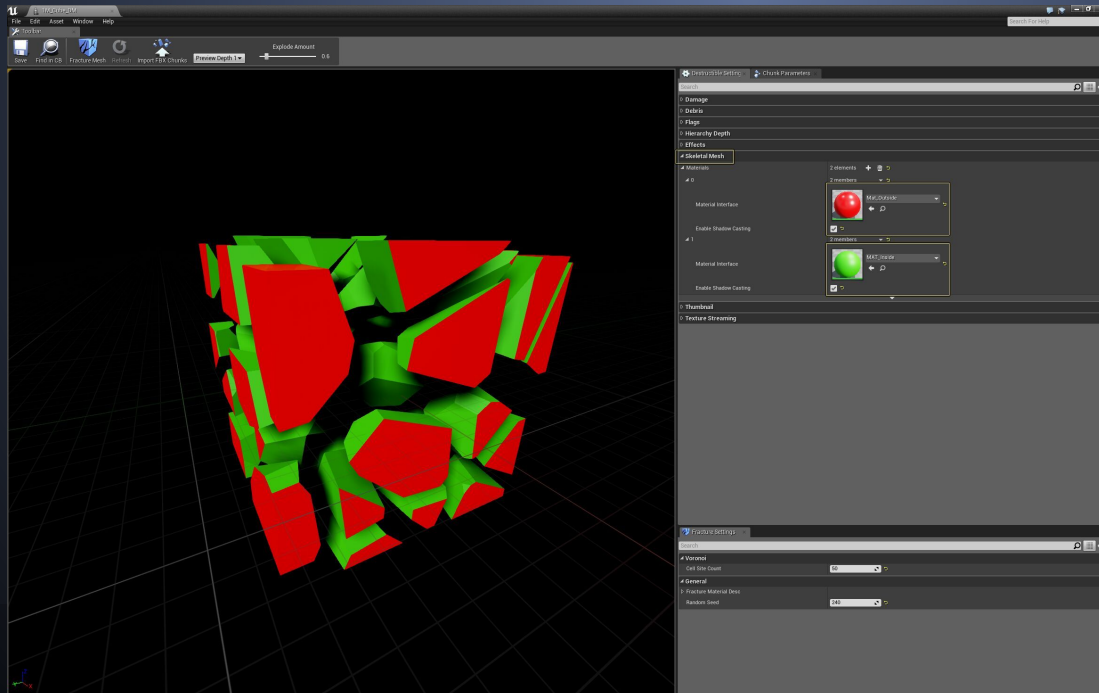


- El componente **Blueprint Spline** sirve para representar estas **curvas definidas a trozos**, y tiene múltiples usos posibles (ej. animar)
- El componente **Blueprint Spline Mesh** es mucho más concreto: sirve para **deformar una malla estática** según un trozo de curva spline entre dos puntos A y B

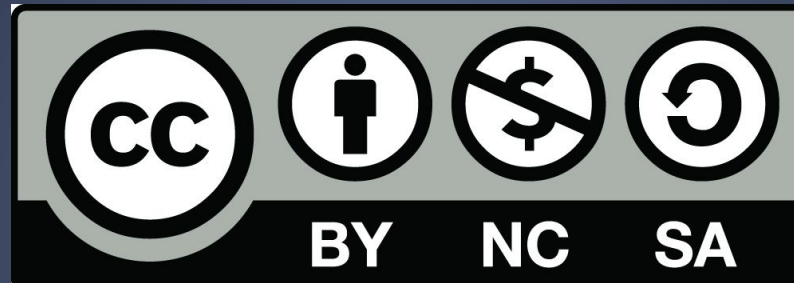


Mallas destruibles

- Una malla estática puede convertirse fácilmente a **malla destruible** con un **plugin**
 - El editor tiene un botón para ver la **fractura**
 - Ahora hace falta un **material externo** y otro **interno**



Críticas, dudas, sugerencias...



** Licencia sólo aplicable al texto original de estas diapositivas*

Federico Peinado (2019-2026)

www.federicopeinado.es