

Les Inputs

La gestion des inputs

```
if(Input.GetKey(KeyCode.LeftArrow))  
{  
    ...  
}
```

ATTENTION : la gestion des inputs doit être faite dans la fonction **Update** sinon vous aurez des comportements... inconsistants



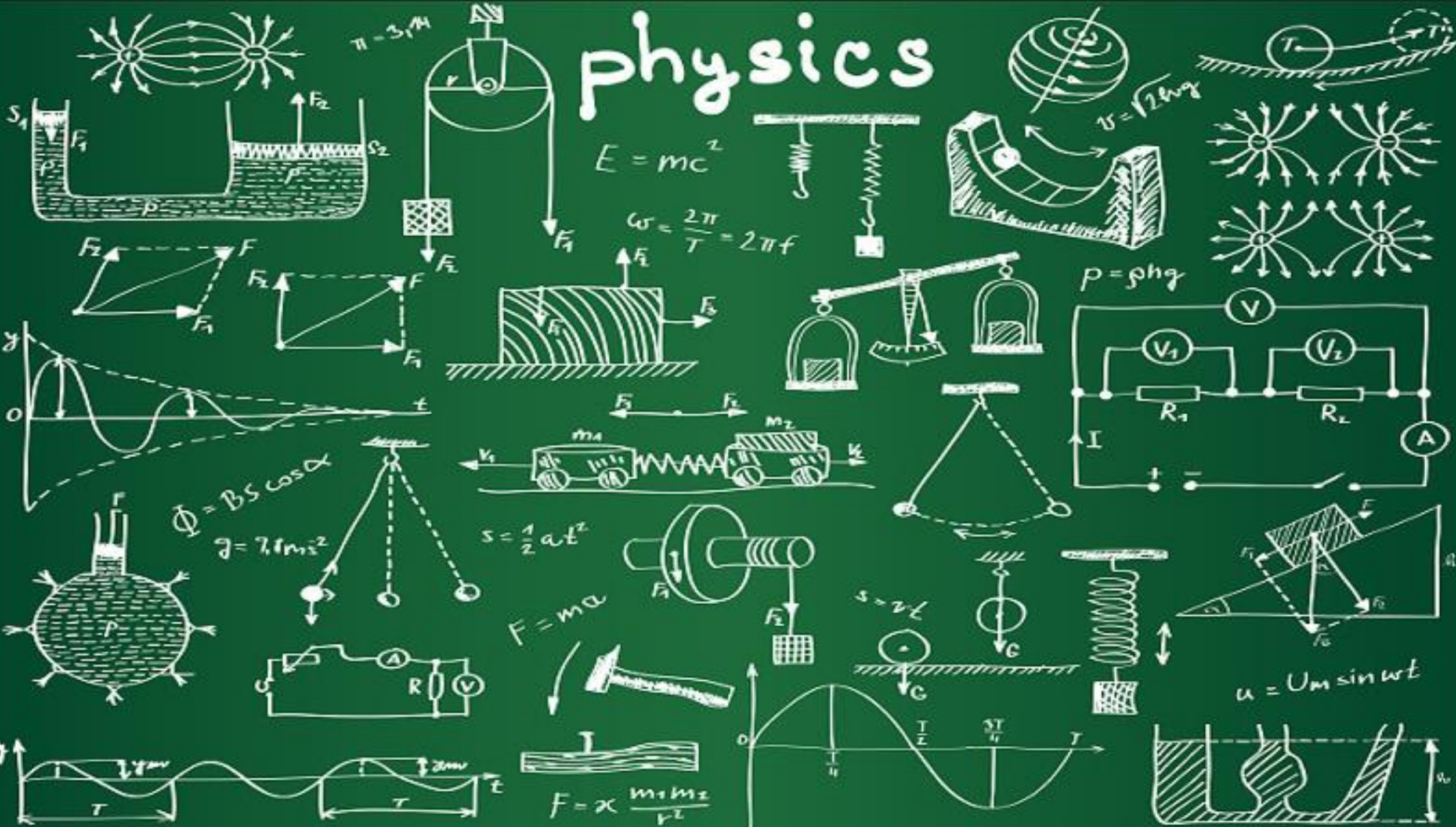
La gestion des inputs

Quelques exemples :

- `GetKey` ➔ Fonction, renvoie vrai tant que la touche est appuyée
- `GetKeyDown` ➔ Fonction, renvoie vrai la première frame où la touche est appuyée
- `GetKeyUp` ➔ Fonction, renvoie vrai la frame où la touche a été relâchée
- `MousePosition` ➔ Propriété, renvoie les coordonnées de la souris à l'écran
- `GetMouseButton` ➔ Fonction, renvoie vrai tant que le bouton de la souris est appuyé

La physique

physics



Comment ça marche ?

Des calculs sont faits rapidement,

(sur la base de lois physiques et des équations complexes)

Pour faire en sorte que ça ait l'air

« vrai »

Comment ça marche ?

Des calculs sont faits **rapidement**,

(sur la base de lois physiques et des équations complexes)

Pour faire en sorte que **ça ait l'air**

« vrai »

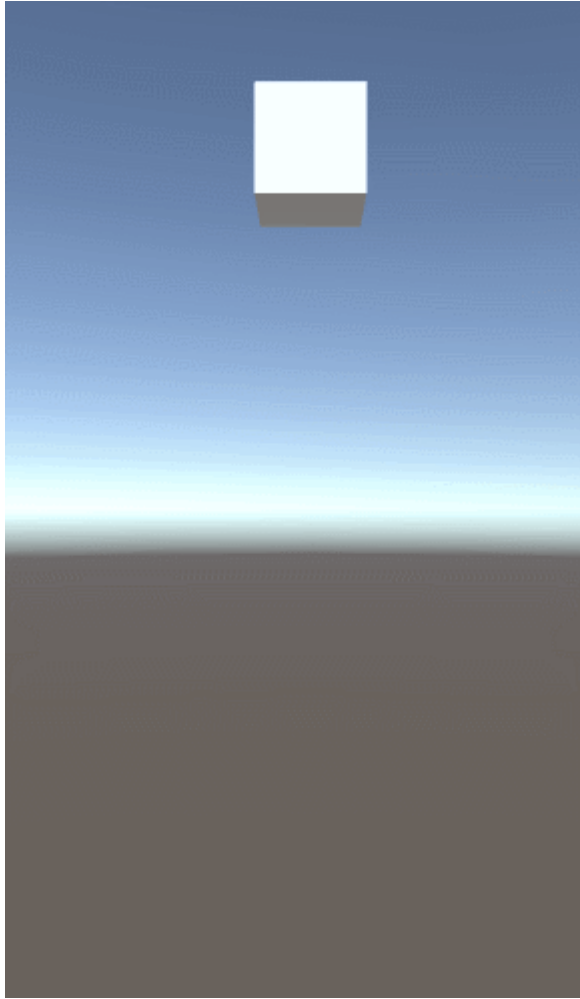
Comment ça marche ?

Ces calculs doivent être faits

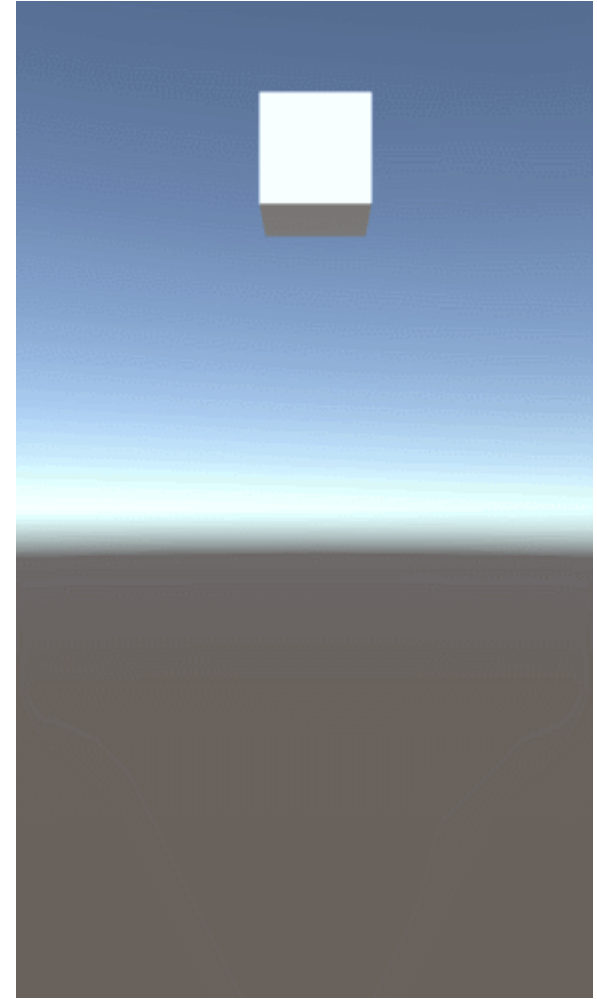
BEAUCOUP de fois

pour obtenir un résultat réaliste

Assez de simulations



Pas assez de simulations



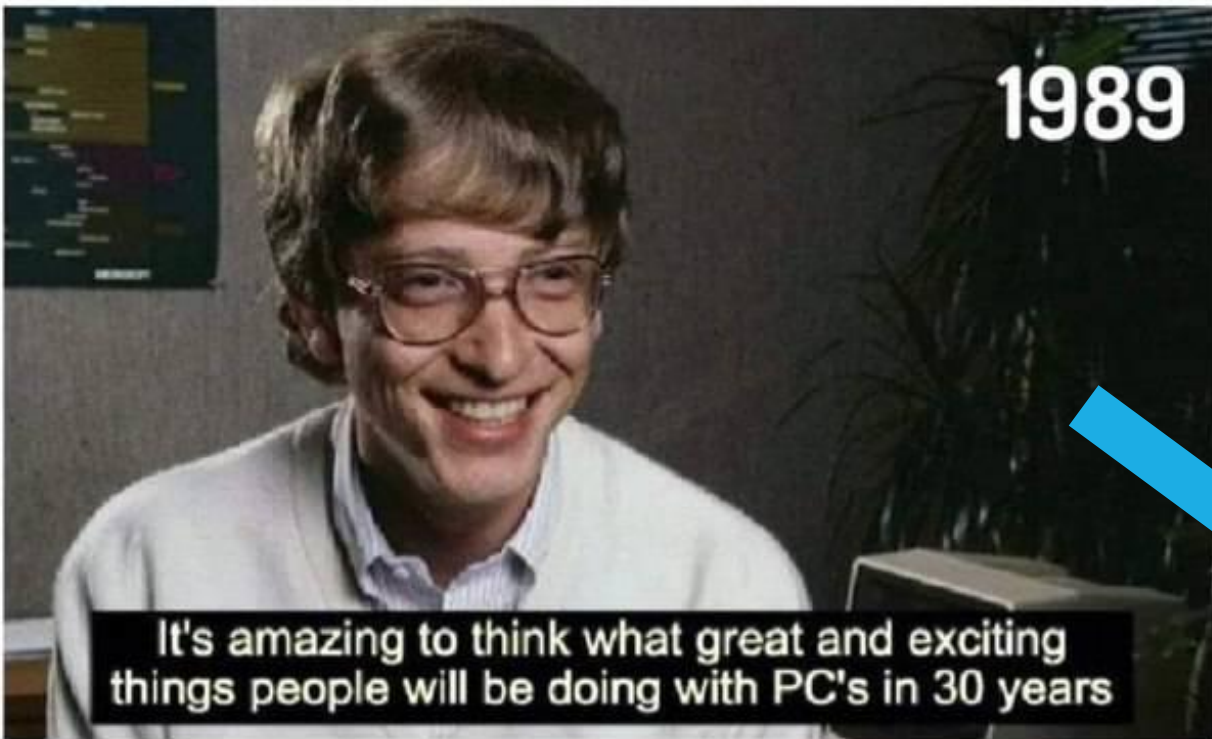
Comment ça marche ?

On n'a qu'à augmenter le nombre de calculs physiques !

Comment ça marche ?

On n'a qu'à augmenter le nombre de calculs physiques !

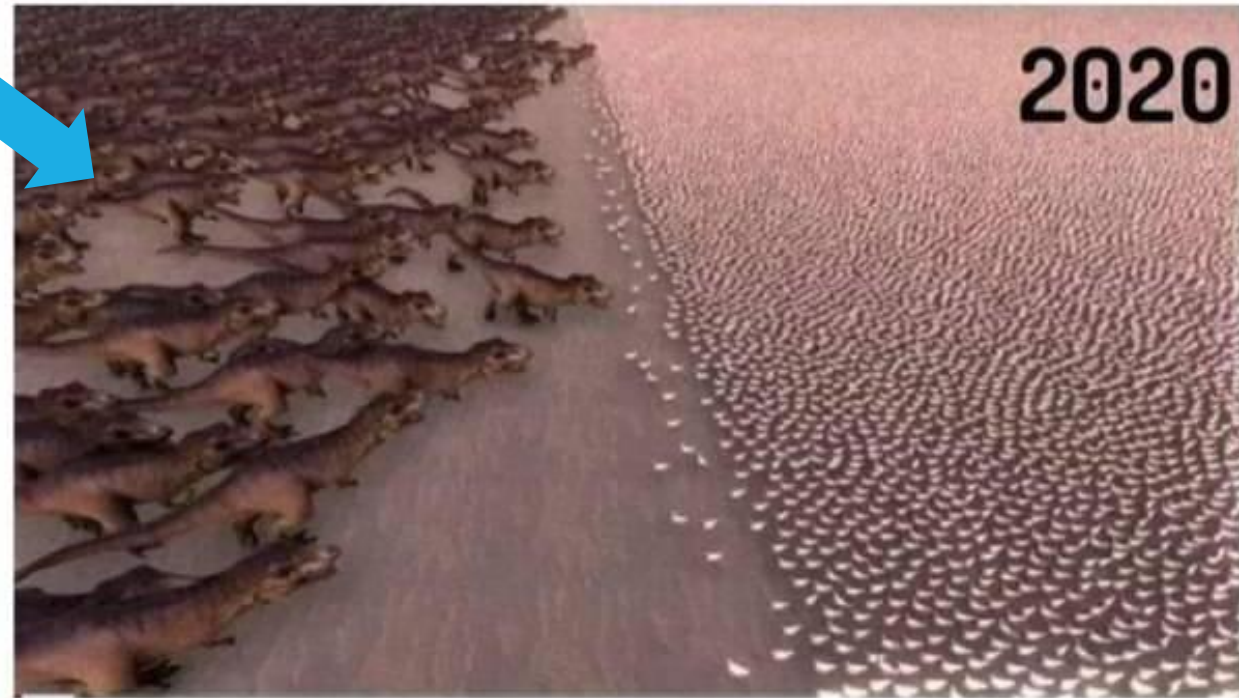




1989

#BillGatesIsProud

#Humanity



2020

1.000 T-REX vs 80.000 CHICKENS -
Ultimate Epic Battle Simulator

Comment ça marche ?

Plus vous ajoutez d'objets physiques à manipuler pour le moteur physique, plus ça va « coûter » cher en temps de calcul

La physique dans Unity

Deux moteurs pour le prix d'un

Unity propose 2 moteurs physiques :

3D

2D

Bonus: Intégration de Havok



Bonus: Intégration de Havok

- Assassin's Creed Odyssey
- The Crew 2
- Dishonored 2
- Far Cry 5
- For Honor
- The Last Guardian
- Forza Horizon 4
- No Man's Sky
- ... et ~600 jeux de plus



Powering physics in games since 2000

**Beaucoup d'échecs
commerciaux**

Deux moteurs pour le prix d'un

- Ces deux moteurs **NE SONT PAS** compatibles
- Faites attention quand vous allez utiliser les composants physiques et les outils de scripting

Deux moteurs pour le prix d'un

- Les Components 3D ont un nom donné :
 - Exemple: BoxCollider
- Les Components 2D ont toujours un suffixe:
 - Exemple: BoxCollider2D

Les concepts principaux

➤ Pour utiliser le moteur physique correctement, il faut utiliser 2 types de composants :

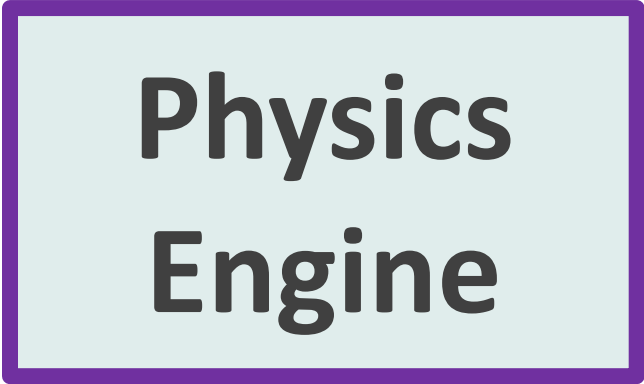
1. Rigidbodies
2. Colliders

Rigidbody

Unity VS Moteur Physique

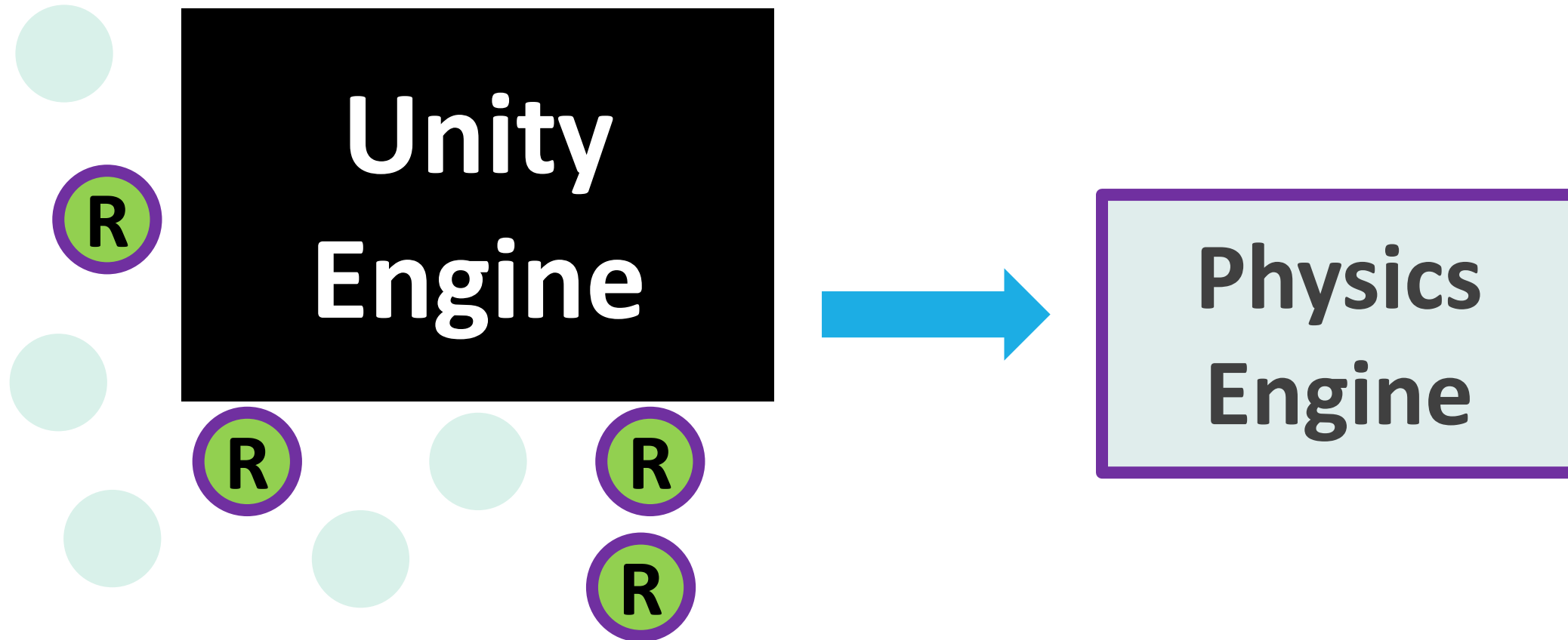


**Unity
Engine**

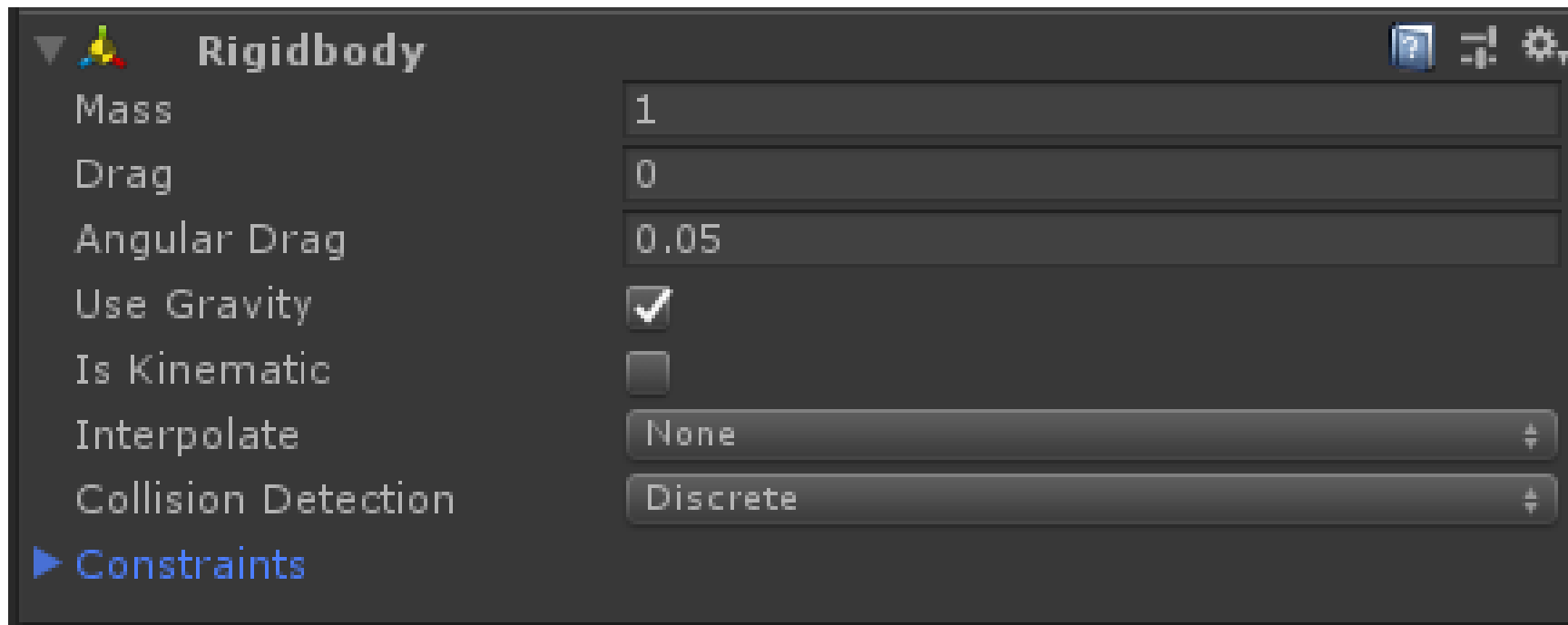


**Physics
Engine**

Unity VS Moteur Physique



Le composant Rigidbody

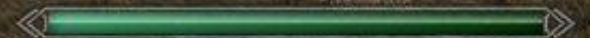




Shut up Jane. I'M FLYING!

Colliders

Stop goofing around Bob
you're scaring the player



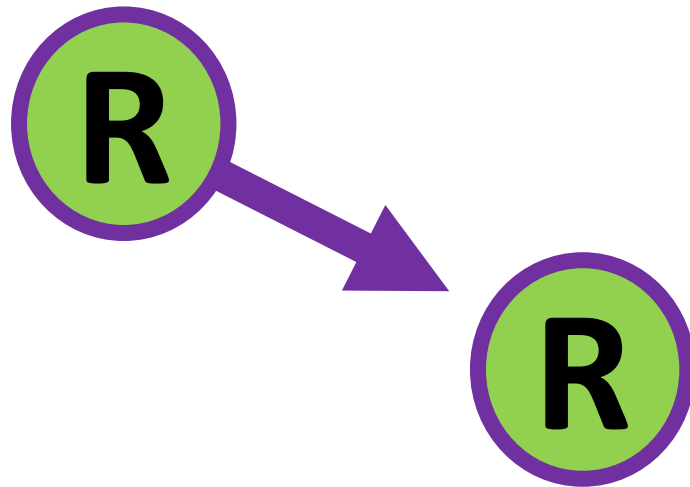
A quoi servent les
Colliders ?

A quoi servent les Colliders ?

Gérer les Collisions*

*Merci captain obvious

A quoi servent les Colliders ?

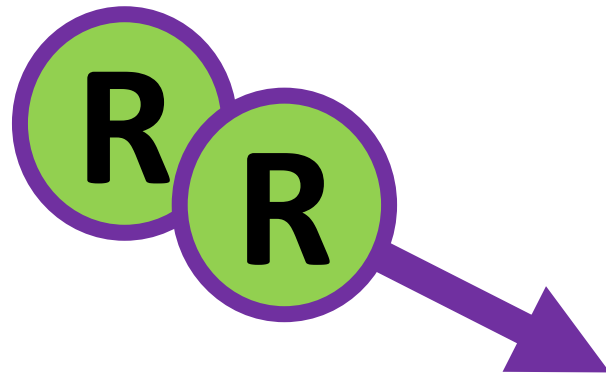


**Physics
Engine**

Le moteur physique
bouge un objet

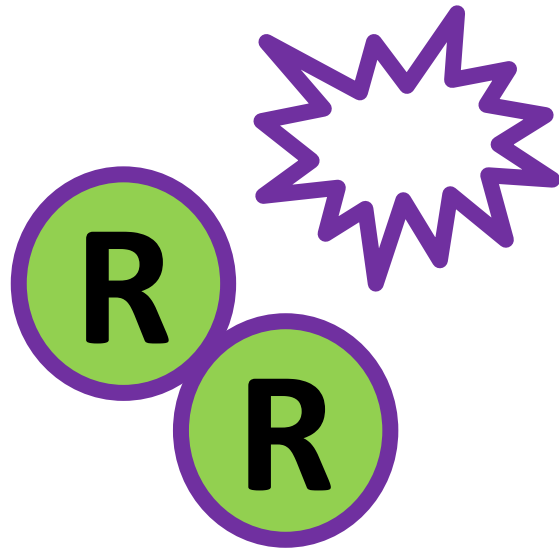
A quoi servent les Colliders ?

Pas de collider, il ne se passe rien. Le premier objet passe à travers le second



**Physics
Engine**

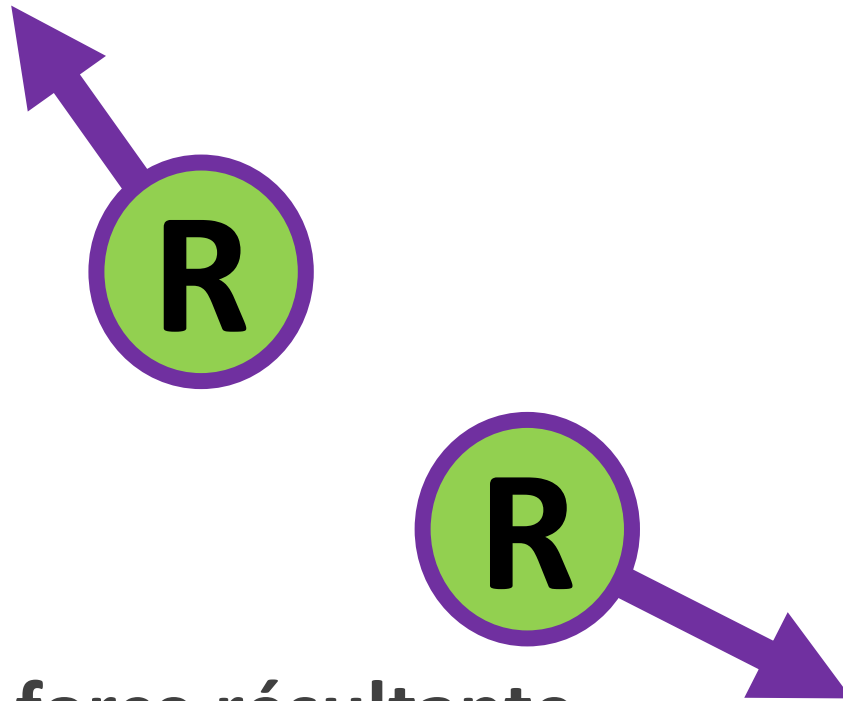
A quoi servent les Colliders ?



**Physics
Engine**

Si ils onot un collider... Là le moteur physique calcule des trucs !

A quoi servent les Colliders ?



**Physics
Engine**

... comme la force résultante
de leur collision

Les composants de type « Colliders »

➤ Il y a plusieurs types de colliders :

➤ BoxCollider

➤ SphereCollider

➤ MeshCollider

➤ CapsuleCollider

➤ ...

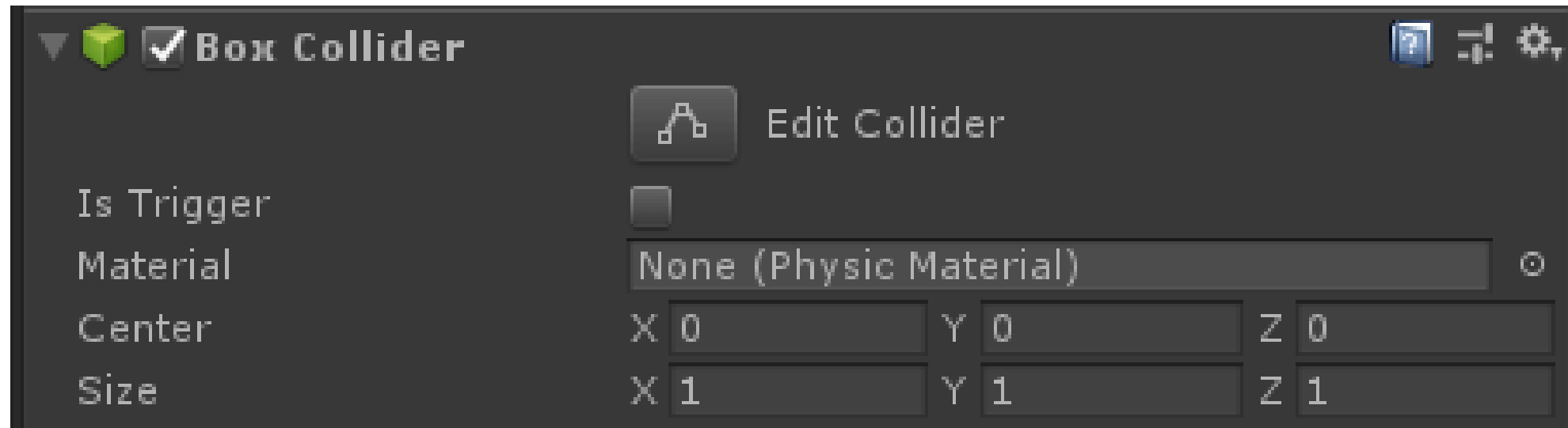
**Toutes les
primitives ont
un collider
associé**

Les modes de collision

Cas d'usage

- Vous voulez créer une porte qui s'ouvre toute seule quand le joueur s'en approche
- Vous voulez démarrer une cut-scene quand le joueur entre dans la salle du boss
- vous voulez déclencher un piège lorsque le joueur marche sur une plaque de pression

Triggers VS Colliders

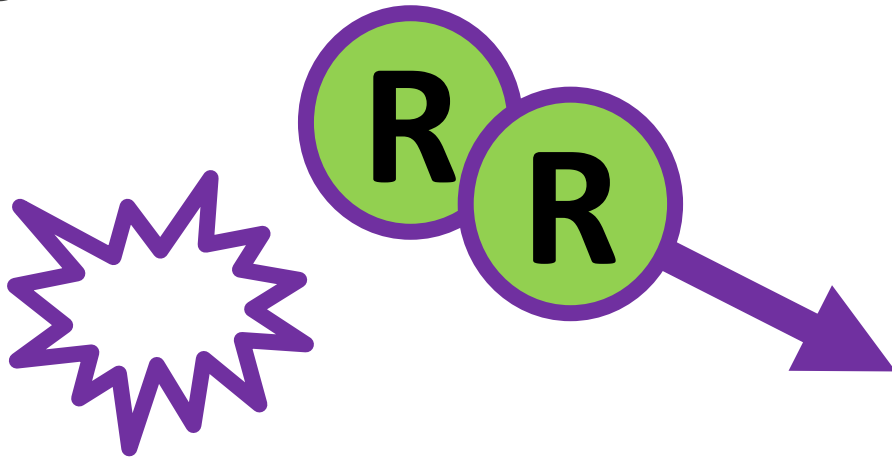


Triggers VS Colliders

Avec un collider marqué comme « trigger »,
le moteur va détecter les collisions mais pas
calculer les forces résultantes

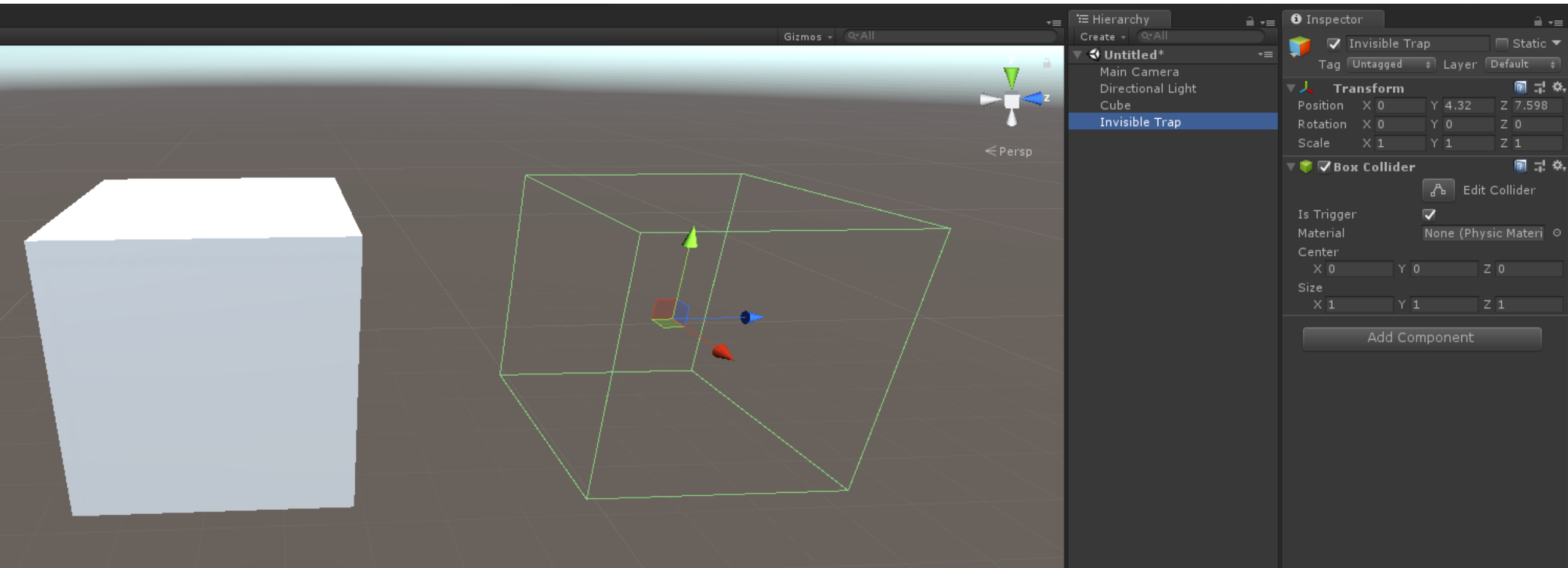
Triggers VS Colliders

Hey Unity ! Il
s'est passé un
truc !



**Physics
Engine**

Un trigger peut être invisible



Réagir aux événements physiques

Les événements physiques

- **Enter** : la collision a été détectée sur cette frame physique
- **Stay**: la collision était détectée à la frame précédente et est toujours détectée sur la frame physique courante
- **Exit**: la collision était détectée à la frame précédente mais n'est plus sur la frame physique courante

Les événements physiques

- Vous pouvez détecter deux familles d'événements :
 - Collisions
 - Triggers

Exemple de code

```
public class CollisionExample : MonoBehaviour
{
    0 references
    public void OnTriggerEnter(Collider other)
    {
        Debug.Log("Something entered the trigger: " + other.name);
    }
}
```

Les événements physiques

COLLISIONS

- OnCollisionEnter
- OnCollisionStay
- OnCollisionExit

TRIGGERS

- OnTriggerEnter
- OnTriggerStay
- OnTriggerExit

Les événements physiques(2D)

COLLISIONS

- OnCollisionEnter2D
- OnCollisionStay2D
- OnCollisionExit2D

TRIGGERS

- OnTriggerEnter2D
- OnTriggerStay2D
- OnTriggerExit2D

Les rayons

C'est quoi un rayon ?

- Un « Rayon », c'est comme un laser qu'on lance sur la scène d'un point de départ dans une direction donnée
- On peut récupérer des informations sur ce qu'il a touché en chemin
- C'est utilisé pour détecter des objets dans une direction

Comment lancer un rayon

- A utiliser préférence dans FixedUpdate
- Il y a plusieurs syntaxes possibles
- Ces deux exemples font la même chose
 - « Raycast » renvoie vrai si le laser touche quelque chose entre la position du transform et son forward

```
if(Physics.Raycast(transform.position, transform.forward))  
{  
    print("Touché !");  
}
```

```
Ray ray = new Ray(transform.position, transform.forward);  
if(Physics.Raycast(ray))  
{  
    print("Touché !");  
}
```

Comment lancer un rayon

- RaycastHit est une structure qui contient des détails sur la détection du raycast
- Renvoie vrai si il touche quelque chose... et met le résultat dans « hit »
- Le mot clé « out » est obligatoire ici pour pouvoir écrire dans la structure

```
RaycastHit hit;  
if(Physics.Raycast(transform.position, transform.forward, out hit))  
{  
    print(hit.collider.name);  
}
```

Le mot de la fin

- Faites **SUPER** attention à propos du nom des fonctions d'événements physique
 - La casse est importante, le type des paramètres aussi
- Dans le doute, vérifiez toujours la documentation :
 - <https://docs.unity3d.com/ScriptReference/MonoBehaviour.OnTriggerEnter.html>

GO
