**5. Collision overview**

1. За что отвечают коллизии, где он у нас у персонажа? Что благодаря ней НЕ происходит?

2. Логические трейсы – что это?

3. Как запустить консоль? Какие есть команды для GPU, FPS и коллизий.

4. Вытащили сферу и куб – какие есть параметры у них в разделе коллизии?

5. Collision Enabled – какие четыре варианта есть?

6. Object type – какие тут есть?

7. Таблица Responses (их там два вида)

8. Какие есть три варианта реакции? Что будет происходит при взаимодействии персонажа с объектами при разных вариантах? Какое есть у SpringArm поле из-за которого мы не можем заглянуть за объект? Как исправить это поведение?

9. Как отслеживать моменты «перекрывания» и столкновения?

10. Как можно создать свои собственные каналы и типы объектов?

1. Коллизии отвечают за физические взаимодействия объектов в мире. В базовом классе Character создается компонент коллизии Capsule. Благодаря ей мы не проваливаемся сквозь пол, прыгаем по ступенькам, врезаемся в другие объекты и не может пройти их насквозь.

2. Благодаря коллизиям мы можем делать логические трейсы – то есть в коде посмотреть, пересекает ли отрезок из точки А в точку Б какой-либо объект на сцене (таким образом мы реализуем стрельбу).

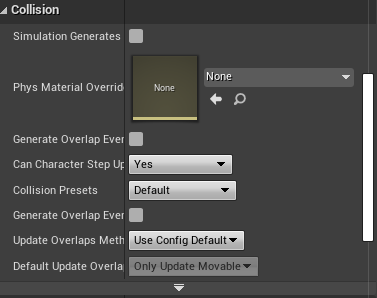
3. В UE есть консоль – запускается с помощью ` (апостроф).

Теперь мы можем вводить различные команды для дебаггинга. Stat FPS – количество FPS, Stat GPU – статистика по GPU. Для скрытия надо либо ввести ту же команду еще раз, либо Stat None.

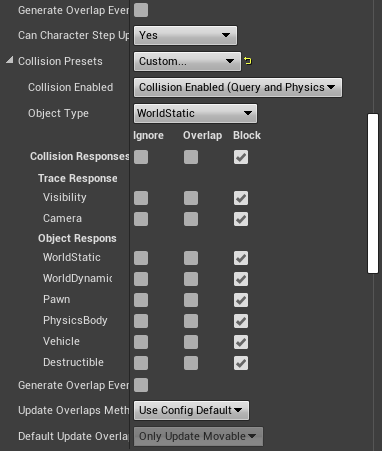
Введем команду Show Collision – она отображает все коллизии на сцене. Компоненты коллизии чисто логические и по умолчанию скрыты.

4. Вытащим на сцену объекты куба и сферы.

В панели Details находим раздел Collision.



Здесь есть проперти Collision Presets – по умолчанию стоит Default. Поменяв на Custom, мы сможем настроить все поведение нашего Collision Preset’a:



5. Первое – Collision Enabled – в данном enum есть 4 варианта:

NoCollision – все коллизии игнорируются.

Query Only (No Physics Collision) – если у тела не включена физическая симуляция, либо нас интересуют только логические трейсы с данным объектом – выбираем этот элемент.

Physics Only (No Query Collision) – если у тела включена физическая симуляция.

Четвертая – Collision Enabled – объединяет 2 и 3.

6. Далее идет поле Object type:

Изображение выглядит как текст

Автоматически созданное описание

7. Далее идет таблица, в которой задается то, каким образом объект данного типа реагирует с объектами других типов (игнорирует, передает сообщение о пересечении или блокирует).

А еще тут есть Trace Response – в ней определены два канала, данные каналы виртуальны и используются для получения информации при логических трейсах.

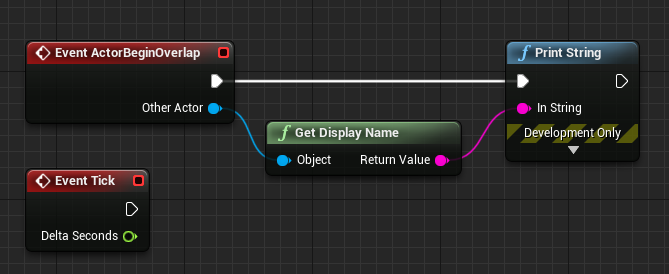
8. Если мы зайдем в BP нашего персонажа, перейдем в компонент капсулы и найдем там меню с коллизиями, по умолчанию стоит Pawn. Выберем Custom. Object type стоит в Pawn. Реакция на все остальные объекты стоит в Block. Понятное дело, что происходит.

Если установить у сферы, например, Overlap, то мы сможем ходить через нее, а через куб нет.

При этом наблюдается интересный эффект при проходе через сферу – камера оказывается в середине персонажа. Это поведение связано со SpringArm, который как раз таки не позволяет заглянуть за стены или что-то в таком роде. За данное поведение отвечает проперти ProbeChannel. То есть наш SpringArm может также взаимодействовать с миром. При этом имя канала коллизии - Camera.

Если мы исправим в TraceResponses у Camera реакцию в Ignore, то такого поведения не будет.

9. У нас так же есть возможность отслеживать момент, когда мы столкнулись с каким-то объектом, либо пересекли его. В ивент графе персонажа есть нода Event ActorBeginOverlap. Если мы выведем сюда имя объектов, которые мы пересекаем, то получим их на экране.



Дополнительно надо у объекта сферы выставить флаг Generate Overlap Events.

Аналогичным образом можно получить информацию о столкновениях – для этого надо активировать флаг Simulation Generates Hit Events. Событие, которое вызывается при столкновениях, называется Event Hit (в блюпринтах).

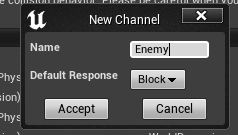
10. Рассмотрим как можно создать свои собственные каналы и типы объектов. Project Settings находим категорию Collision. Имеется три категории – Object Channels, Trace Channels и Preset.

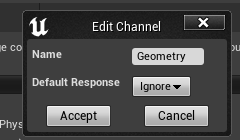
Мы можем создать свой пресет и определить для него параметры:



Можем менять и существующие пресеты.

11. Создадим собственный канал объекта (Object Channel):





То есть все объекты будут игнорировать последний тип объекта по умолчанию.

При этом в Object Type стали доступны два этих типа:

Изображение выглядит как текст

Автоматически созданное описание

Если мы зайдем к нашему персонажу в раздел коллизий, мы увидим, что у него появились эти Object Response и с теми настройками, которые мы указали.

