**Extracting the carrying object**

We will create an actor with collision box to extract.

First create an actor called FPSExtraction

Then create a box collision in it through C++.

Add following to the Header file (Under protected).

***UPROPERTY(VisibleAnywhere, Category = "Components")***

***UBoxComponent\* OverlapComp;***

forward decleration required.

***class UBoxComponent;***

Then go to the C++ file and add include file

***#include "Components/BoxComponent.h"***

in constructor add the following

***OverlapComp = CreateDefaultSubobject<UBoxComponent>(TEXT("OverlapComp"));***

***OverlapComp->SetCollisionEnabled(ECollisionEnabled::QueryOnly);***

***OverlapComp->SetCollisionResponseToAllChannels(ECR\_Ignore);***

***OverlapComp->SetCollisionResponseToChannel(ECC\_Pawn, ECR\_Overlap);***

***OverlapComp->SetBoxExtent(FVector(200.0f));***

***RootComponent = OverlapComp;***

**Dynamic Delegates**

Now we need an event for overlap detection.

We could use ActorNotifyBeginOverlap like in PickupActor

But we will use an alternative

We are going to bind a new function specifically to 'OnComponentBeginOverlap' method.

For this we use a feature called DynamicDelegates

***OverlapComp->OnComponentBeginOverlap.AddDynamics(this, &AFPSExtractionZone::HandleOverlap);***

Intellisence will not work because HandleOverlap function is not declared.

So declare the function in header file as follows.

***UFUNCTION()***

***void HandleOverlap();***

Function requires parameters matching to OnComponentBeginOverlap.

To get it right click on 'OnComponentBeginOverlap' and go to definition.

You will be taken to 'PremitiveComponent.h' to the following line of code

FComponentBeginOverlapSignature OnComponentBeginOverlap;

Right click on 'FComponentBeginOverlapSignature' and go to definition. (or search)

You will be taken to the following line in the same file.

DECLARE\_DYNAMIC\_MULTICAST\_DELEGATE\_SixParams( FComponentBeginOverlapSignature, UPrimitiveComponent\*, OverlappedComponent, AActor\*, OtherActor, UPrimitiveComponent\*, OtherComp, int32, OtherBodyIndex, bool, bFromSweep, const FHitResult &, SweepResult);

Copy the parameters and paste it in the decleration of HandleOverlap function (remove the commas)

The final HandleOverlap function should look like this.

***UFUNCTION()***

***void HandleOverlap(UPrimitiveComponent\* OverlappedComponent, AActor\* OtherActor, UPrimitiveComponent\* OtherComp, int32 OtherBodyIndex, bool bFromSweep, const FHitResult & SweepResult);***

Create Definition for HandleOverlap and add UE\_LOG

***void AFPSExtractionZone::HandleOverlap(UPrimitiveComponent \* OverlappedComponent, AActor \* OtherActor, UPrimitiveComponent \* OtherComp, int32 OtherBodyIndex, bool bFromSweep, const FHitResult & SweepResult)***

***{***

***UE\_LOG(LogTemp, Log, TEXT("Overlapped with extraction zone"));***

***}***

To get a visual representation add following

***OverlapComp->SetHiddenInGame(false);***

In the Editor add the FPSExtractionBox in the world and play and go over it to see the message in output log.

**References**

<https://docs.unrealengine.com/en-US/Programming/UnrealArchitecture/Delegates/index.html>

<https://docs.unrealengine.com/en-US/Programming/UnrealArchitecture/Delegates/Dynamic/index.html>