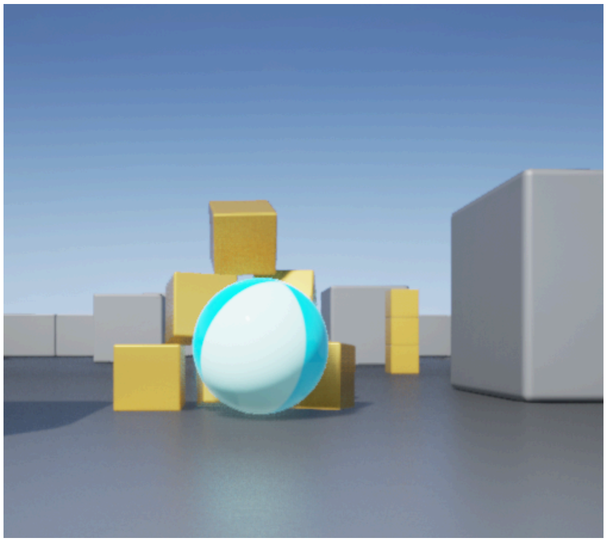
Developer
/ Documentation
/ Unreal Engine
/ Unreal Engine 5.4 Documentation
/ Making Interactive Experiences
/ Gameplay Framework
/ Pawn

# Pawn

Overview of Pawns



The **Pawn** class is the base class of all Actors that can be controlled by players or AI. A Pawn is the physical representation of a player or AI entity within the world. This not only means that the Pawn determines what the player or AI entity looks like visually, but also how it interacts with the world in terms of collisions and other physical interactions. This can be confusing in certain circumstances as some types of games may not have a visible player mesh or avatar within the game. Regardless, the Pawn still represents the physical location, rotation, etc. of a player or entity within the game. A Character is a special type of Pawn that has the ability to walk around.

By default, there is a one-to-one relationship between Controllers and Pawns; meaning, each Controller controls only one Pawn at any given time. Also, Pawns spawned during gameplay are not automatically possessed by a Controller.

In **Blueprints**, the best way to add movement to your Pawn-derived class is with **SetActorLocation**. With SetActorLocation, you can decide if you want to teleport to the location, or sweep. When sweeping, your Pawn will move along the direction and stop if it hits something.

# Default Pawn

While the Pawn class provides only the essentials for creating a physical representation of a player or AI entity in the world, the *DefaultPawn* subclass comes with some additional Components and functionality.

The DefaultPawn class contains a native *DefaultPawnMovementComponent*, a spherical *CollisionComponent*, and a *StaticMeshComponent*. To control the *DefaultPawnMovementComponent*, as well as the camera, a Boolean to add default movement bindings is also present in the DefaultPawn class, and is set to *true* by default.

## DefaultPawnMovementComponent

The movement style set for the *DefaultPawnMovementComponent* is a no-gravity, flying style. In addition to the usual *MovementComponent* variables, it includes `MaxSpeed`, `Acceleration`, and `Deceleration` float values. These three variables are also accessible in DefaultPawn-derived Blueprints.

# Spectator Pawn

The *SpectatorPawn* class is a subclass of DefaultPawn. Through a GameMode, different classes can be specified as defaults for Pawn and SpectatorPawn, and this class provides a simple framework ideal for spectating functionality. Like DefaultPawn, it has a spherical *CollisionComponent*, although the *StaticMeshComponent* is not created due to the implementation of `.DoNotCreateDefaultSubobject(Super::MeshComponentName)`. The movement for the SpectatorPawn class is handled in the

*SpectatorPawnMovementComponent*; the no-gravity flying behavior is the same as in the *DefaultPawnMovementComponent*, with added code for handling or ignoring time dilation as necessary.