Developer
/ Documentation
/ Unreal Engine
/ Unreal Engine 5.4 Documentation
/ Making Interactive Experiences
/ Gameplay Targeting System

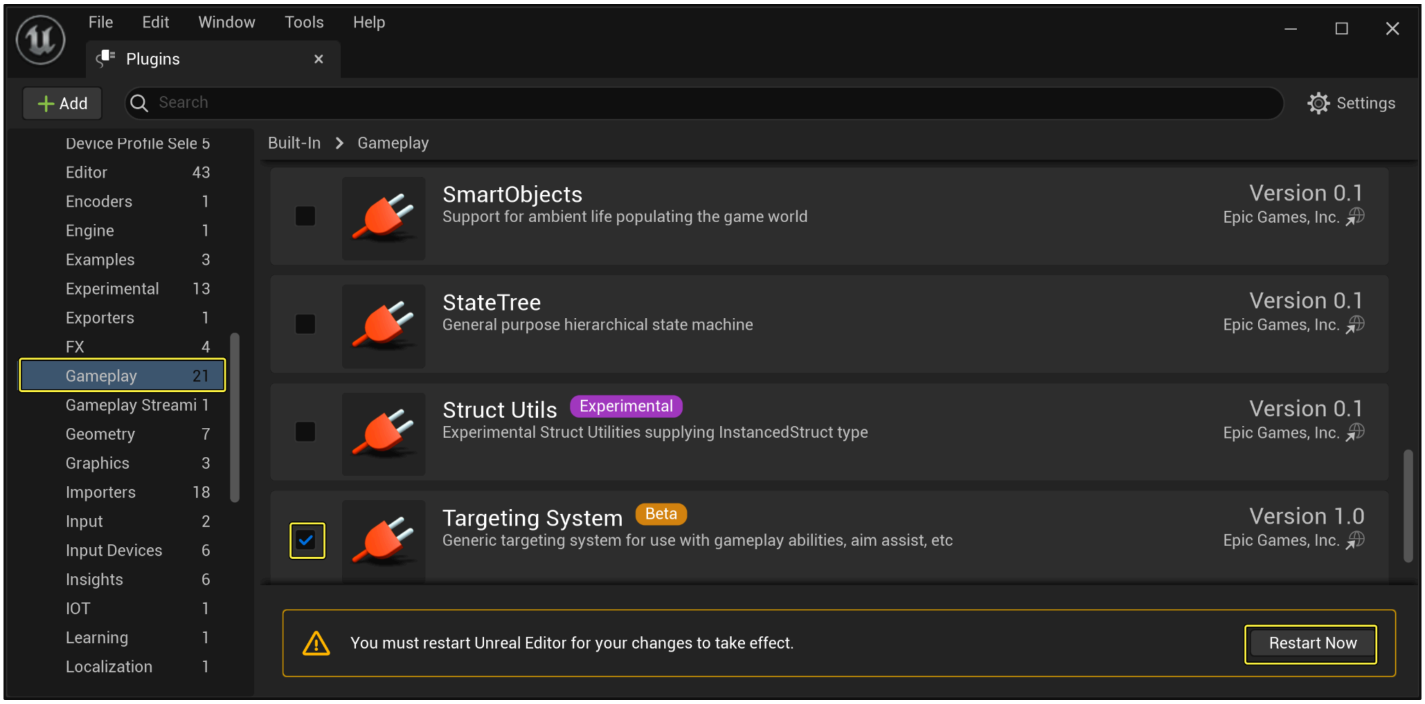
Gameplay Targeting System

An overview of the Gameplay Targeting System plugin framework in Unreal Engine.

Gameplay Targeting System

The Gameplay Targeting System is a way to create data-driven targeting requests. It expands on the Gameplay Ability System Plugin, but could be used outside of the feature.

You can enable the Targeting System from the Editor by navigating to **Edit** > **Plugins** > **Gameplay** > **Gameplay Abilities** > **Targeting System**.



Device Profile Sele	5
Editor	43
Encoders	1
Engine	1
Examples	3
Experimental	13
Exporters	1
FX	4
Gameplay	21
Gameplay Streami	1
Geometry	7
Graphics	3
Importers	18
Input	2
Input Devices	6
Insights	6
IOT	1
Learning	1
Localization	1

□ SmartObjects — Support for ambient life populating the game world — Version 0.1 — Epic Games, Inc.

□ StateTree — General purpose hierarchical state machine — Version 0.1 — Epic Games, Inc.

□ Struct Utils (Experimental) — Experimental Struct Utilities supplying InstancedStruct type — Version 0.1 — Epic Games, Inc.

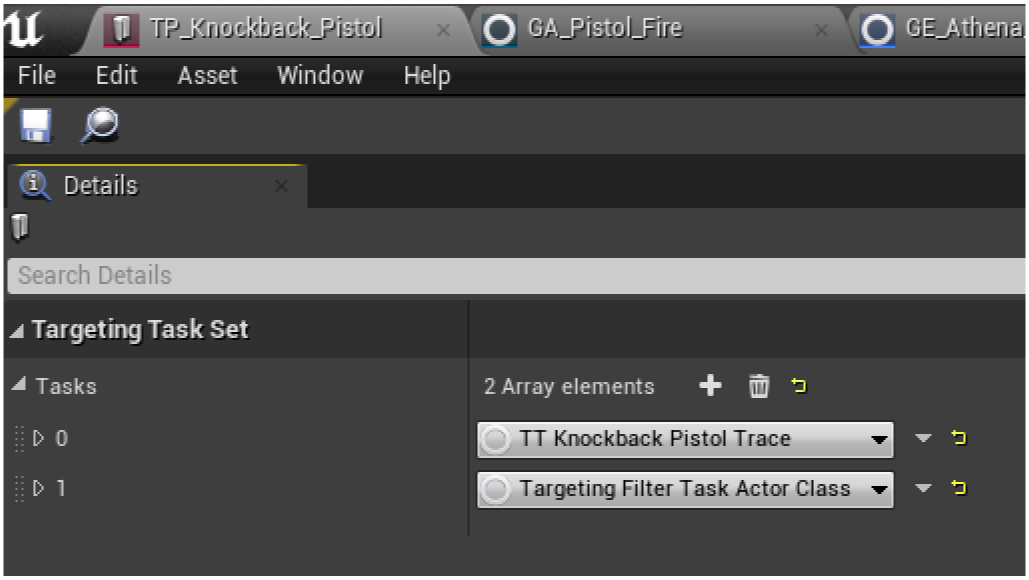
☑ Targeting System (Beta) — Generic targeting system for use with gameplay abilities, aim assist, etc — Version 1.0 — Epic Games, Inc.

You must restart Unreal Editor for your changes to take effect.

Setting up the Targeting System Plugin.

Data and Design

The Targeting System uses a **Targeting Preset**. This Targeting Preset is a data asset that defines a set of **Targeting Tasks** that are run from top to bottom for an executed targeting request.



Targeting Task Set

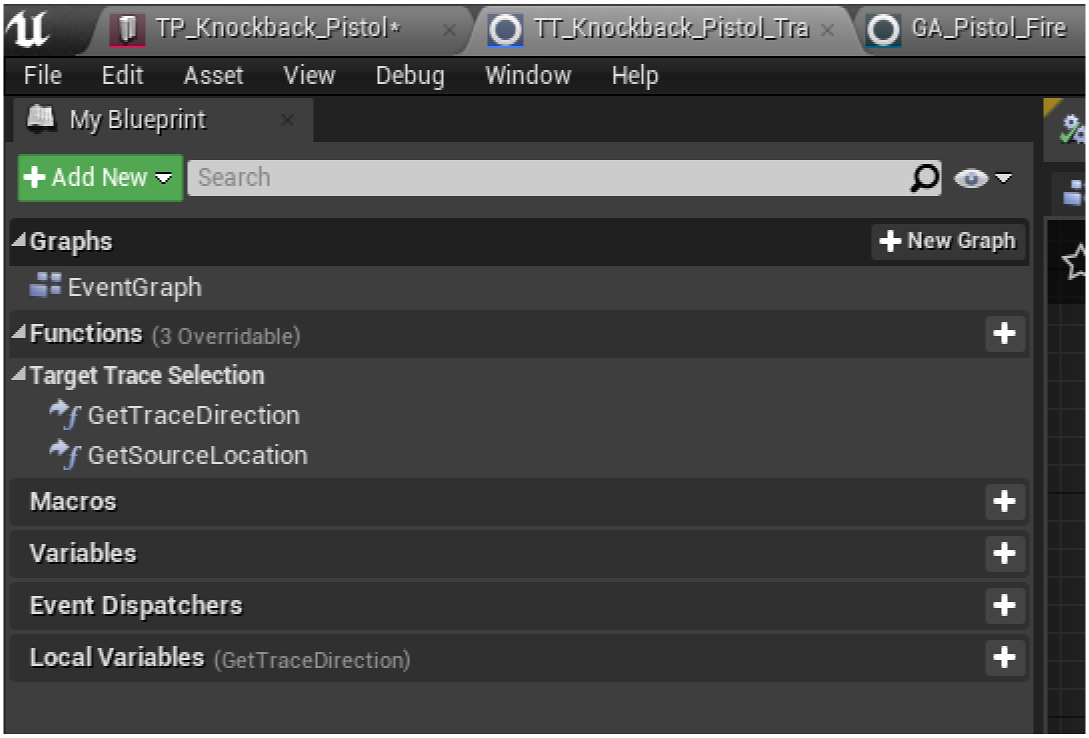
Tasks — 2 Array elements

0 — TT Knockback Pistol Trace

1 — Targeting Filter Task Actor Class

The TP_Knockback_Pistol is a targeting preset set up with a Trace targeting task and Filter by the Actor class targeting task.

A feature of targeting presets is that each task supports inline property editing. You can define reusable targeting tasks and presets that set up how the tasks behave without defining a `UAsset` for each one.

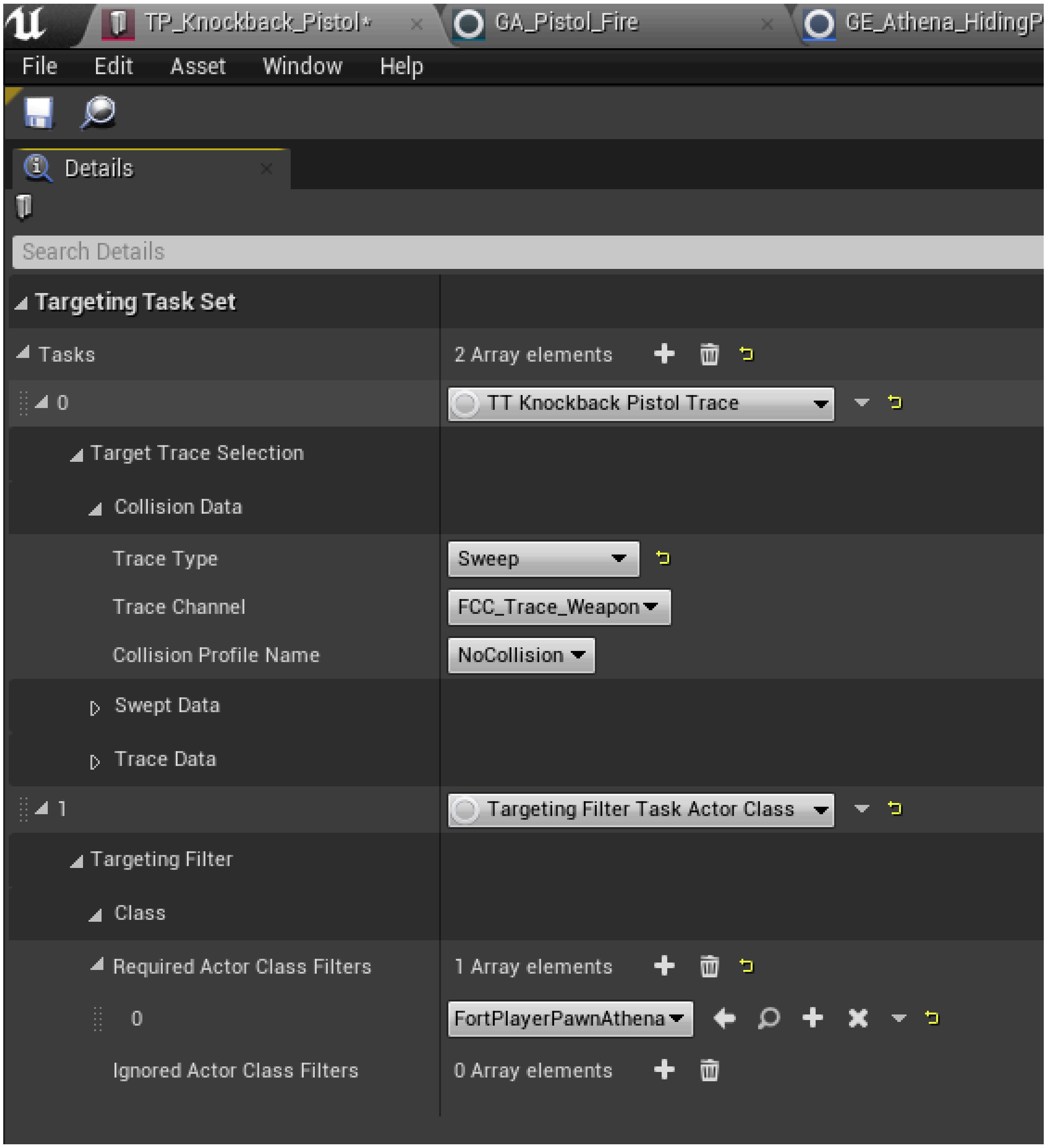


Between the two images above, the Trace Types and the Required Actor Classes are different. This is done in the same targeting preset without having to pick different targeting tasks.

TargetingTask

A **Targeting Task** is a small unit operation meant to perform a desired action. Targeting Tasks can be generally thought about in 3 categories, Selection, Filtering,and Sorting.

Targeting tasks have Blueprint support. Depending on how you engineered your tasks, it can be as simple as setting up Class default properties to handling Blueprint callable events.



Targeting Task Set	
Tasks	2 Array elements
0	TT Knockback Pistol Trace
Target Trace Selection	
Collision Data	
Trace Type	Sweep
Trace Channel	FCC_Trace_Weapon
Collision Profile Name	NoCollision
Swept Data	
Trace Data	
1	Targeting Filter Task Actor Class
Targeting Filter	
Class	
Required Actor Class Filters	1 Array elements
0	FortPlayerPawnAthena
Ignored Actor Class Filters	0 Array elements

Example: The **Knockback Pistol Trace** task is set up with two overridable events to calculate the source location and the direction of the trace. In this example, if you look at the Blueprint code you'll see it gets the pawn and calls a function to get the camera location and camera direction.

Immediate Targeting Requests

Immediate Targeting Requests are performed by calling the `UTargetingSubsystem::ExecuteTargetingRequest` function. The Targeting Request is based on a `UTargetingPreset` Object and is executed immediately, therefore blocking the Game Thread until it is completed.

Async Targeting Requests

Async Targeting Requests are performed by calling the `UTargetingSubsystem::StartAsyncTargetingRequest` function. The Targeting request is based on a `UTargetingPreset` Object and starts immediately, however, it may not block the Game Thread until it is completed.

These requests are useful when a Targeting Task may need to wait for a slow operation (for example, waiting for the server to authenticate a target) or for a gameplay event to occur before targeting can be completed.

> ⓘ All Targeting Tasks defined in the GameplayTargetingSystem Plugin will provide implementations for both Immediate and Asynchronous targeting requests, however, the game code is not enforced to implement both if the game does not use the async functionality.

Async Action - Perform Targeting / Perform Filtering

UAsyncAction_PerformTargeting provides generic Blueprint support for starting an Async Targeting Request or Filtering Action.

Debugging

For a complete list of debug console commands and descriptions, see the Gameplay Targeting Debugging documentation.

Developer Reference

For an in-depth Engineering reference guide, see the Gameplay Targeting Plugin Reference documentation.