



HOW TO MOD PLACEHOLDERS

Prerequisites

- Intro to actor & placeholder mods
- How to patch vanilla actors
- How to delete vanilla actors
- How to add vanilla actors

Introduction

In this guide, we'll look at placeholder patching, deletion, and addition.

WARNING

This guide focuses on the specifics of modding placeholder actors, skipping most descriptions and details of the shared mechanisms covered in the prerequisite docs.

Design & work principles

The main point to understand is that, specifically in Zone Kit, any modifications to placeholders work alongside the other basic actor modification systems. However, in PIE sessions or while in-game, the changes are driven purely by .cfg files. This means that, for example, placeholder patching is more limited and has more restrictions than actor patching.

WARNING

Please refer to the [Known issues & limitations](#) section at the end of the guide for more details.

Let's look at an example of a Spawn Actor Prototype .cfg file generated by a modded placeholder:

```
// BP_TriggerPlaceholder_C_UAID_2CF05D98D6D822BF01_1573067849
6A6C54084935F091EB1102AA665E7EAE : struct.begin
    SID = 6A6C54084935F091EB1102AA665E7EAE
    NodePrototypeVersion = 10
    SpawnOnStart = false
    PositionX = 420528.317809
    PositionY = 193365.749449
    PositionZ = -4266.372202
    RotatorAngleYaw = 79.999999
    RotatorAnglePitch = 0.0
    RotatorAngleRoll = 0.0
    ScaleX = 131.824696
    ScaleY = 162.5
    ScaleZ = 150.0
    TextAboveActor =
    DLC = None
    LevelName = WorldMap_WP/E16_MQ01_C03_LogicLevel_WP
    PlaceholderActorGuid = 2XZX0RTF3C5BRP8HFHMDS7R
    PlaceholderMapPath =
/Game/_Stalker_2/maps/_Stalker2_WorldMap/QuestLevels/E16_MQ01_C03_Logic
Level_WP
    SpawnType = ESpawnType::Trigger
    TriggerShape = ETriggerShape::Box
    NodePrototypeAppliedMods : struct.begin {bpatch}
        MyActorDeletion : struct.begin
            NodePrototypeModVersion = 0
        struct.end
    struct.end
struct.end
```

While different types of Spawn Actor Prototypes have different properties, the following ones are worth going through separately:

1. **SpawnOnStart** – indicates whether the actor will be automatically spawned at the start of the game (either by loading into an existing campaign or starting a new one) or by a specific script or event.

1. Examples of the former type are mostly preplaced items, containers with loot, NPCs, and mutants in the open world.
2. Examples of the latter type are mostly quest-related entities, which need to be spawned manually based on specific conditions, e.g., after starting a quest.
2. SpawnType – the actual type of the Spawn Actor Prototype.
3. NodePrototypeAppliedMods – not present in vanilla placeholders. Explored more in the section below.

NOTE

As with regular .cfg mods, you can tweak the generated Spawn Actor Prototype file yourself (or even create one manually).

In Zone Kit, Spawn Actor Prototype .cfg files are generated at the following locations:

Use case	Definition	Zone Kit path example
Modded placeholder	Stalker2\Mods\<ModName>\Content\GameLite\GameData\SpawnActorPrototypes\...	Stalker2\Mods\MyPlaceholderDeletion\Content\GameLite\GameData\SpawnActorPrototypes\WorldMap_WP\9A41EE1649839E16066221BAE266390E.cfg
New placeholder	Stalker2\Mods\<ModName>\Content\GameLite\ModGameData\SpawnActorPrototypes\.. ..	Stalker2\Mods\MyNewPlaceholder\Content\GameLite\ModGameData\SpawnActorPrototypes\WorldMap_WP\9A41EE1649839E16066221BAE266390E.cfg

It's also important to understand the basics of how placeholder (or, more properly, Spawn Actor Prototype) modifications are actually applied under the hood. Here's an overview of the algorithm, which is run during gameplay:

1. If the corresponding prototype has already been spawned and it has been changed – modded, the version is different, etc. – delete it to prevent duplicates.
2. If it's a SpawnOnStart prototype, spawn a new actor from the modified Spawn Actor Prototype.
3. If it's not a SpawnOnStart prototype, spawn it regularly when needed.

This means that already created actors do not get directly overwritten by mods, but instead the older version is deleted and a new one is recreated using the modded prototype .cfg file.

NOTE

The steps in the algorithm are true for any placeholder modifications – addition, deletion, or patching.

Patching vanilla placeholders

The actor patching system generates not only regular patches when patching placeholders, which are saved to the corresponding Mod Actor Patches Data Asset file, but it also automatically creates a .cfg file based on the modified placeholder:

```
[2026.06.03-17.01.01:260][825]LogModKitPlaceholderManager: Saved modded placeholder config node:
Placeholder SID - 9A41EE1649839E16066221BAE266390E
Config path - ../../Mods/MyActorDeletion/Content/GameLite/GameData/SpawnActorPrototypes/WorldMap_WP/9A41EE1649839E16066221BAE266390E.cfg
```

Here's a comparison table between a vanilla and a patched placeholder prototype .cfg file (in this example, we only changed the location of the placeholder):

Vanilla .cfg	Modded .cfg
<pre>// BP_ItemPlaceholder_C_UAID_00D861D9738D F7D901_1884212591 9A41EE1649839E16066221BAE266390E : struct.begin SID = 9A41EE1649839E16066221BAE266390E SpawnOnStart = true PositionX = 361235.690158 PositionY = 588617.617694 PositionZ = 718.041036 RotatorAngleYaw = -29.139012 RotatorAnglePitch = 61.287165 RotatorAngleRoll = -75.778038 ScaleX = 1.0 ScaleY = 1.0 ScaleZ = 1.0 TextAboveActor = DLC = None LevelName = WorldMap_WP PlaceholderActorGuid = 8D75MTEG5Q7D8H5X88GHY042P PlaceholderMapPath = /Game/_Stalker_2/maps/_Stalker2_WorldM ap/WorldMap_WP MeshPath = StaticMesh'/Game/_Stalker_2/items/cons umable/SM_cns_cossacks_vodka_01_a.SM_c ns_cossacks_vodka_01_a' SpawnedPrototypeSID = Vodka SpawnType = ESpawnType::Item ItemSID = Vodka DisablePhysicsAndCollision = false struct.end</pre>	<pre>// BP_ItemPlaceholder_C_UAID_00D861D97 38DF7D901_1884212591 9A41EE1649839E16066221BAE266390E : struct.begin {bpatch} SID = 9A41EE1649839E16066221BAE266390E SpawnOnStart = true PositionX = 361435.690158 PositionY = 588617.617694 PositionZ = 798.041036 RotatorAngleYaw = -29.139012 RotatorAnglePitch = 61.287165 RotatorAngleRoll = -75.778038 ScaleX = 1.0 ScaleY = 1.0 ScaleZ = 1.0 TextAboveActor = DLC = BaseGame LevelName = WorldMap_WP PlaceholderActorGuid = 8D75MTEG5Q7D8H5X88GHY042P PlaceholderMapPath = /Game/_Stalker_2/maps/_Stalker2_Wor ldMap/WorldMap_WP MeshPath = StaticMesh'/Game/_Stalker_2/items/c onsumable/SM_cns_cossacks_vodka_01_ a.SM_cns_cossacks_vodka_01_a' SpawnedPrototypeSID = Vodka SpawnType = ESpawnType::Item ItemSID = Vodka DisablePhysicsAndCollision = false NodePrototypeAppliedMods :</pre>

```

struct.begin {bpatch}
    MyActorDeletion :
struct.begin

NodePrototypeModVersion = 0
    struct.end
    struct.end
struct.end

```

NOTE

The modded .cfg file has:

- A {bpatch} keyword automatically applied
- A NodePrototypeAppliedMods node containing metadata, which is shared between mods and saved in the save file

Deleting vanilla placeholders

The actor deletion system generates not only regular deletion infos when deleting placeholders, which are saved to the corresponding Mod Actor Deletions Data Asset file, but it also automatically creates a .cfg file based on the modified placeholder:

```

[2026.06.03-17.34.40:532][525]LogModKitPlaceholderManager: Saved placeholder deletion patch:
Placeholder SID - 9A41EE1649839E16066221BAE266390E
Config path - ../../Mods/MyActorDeletion/Content/GameLite/GameData/SpawnActorPrototypes/WorldMap_WP/9A41EE1649839E16066221BAE266390E.cfg

```

Here's an example of a deleted placeholder prototype .cfg file:

```

// BP_ItemPlaceholder_C_UAID_00D861D9738DF7D901_1884212591
9A41EE1649839E16066221BAE266390E : removenode

```

Adding new placeholders

When adding new placeholders, a corresponding Spawn Actor Prototype .cfg file is generated alongside it.

Known issues & limitations

1. Here's a list of placeholders that require starting a new campaign to be properly modified:
 - a. PackOfItems