

# DRG Modding Guide

By Rauliken

(Red color means important, blue is a link inside the document, green is an external link)

(You can click blue text to go to that part, there's also [blue links](#) through the guide)

(You can also go back to the index by clicking "Back to Index" on the right when you see it)

## INDEX

<b>1. Introduction and tools</b>	<b>2</b>
1.1 The Basic Knowledge	2
1.2 The Basic Tools	3
<b>2. Things to learn before modding</b>	<b>6</b>
2.1 Introduction to DRG Packer	6
2.2 Unpacking the game's files	8
2.3 Checking the content of the files	9
2.4 Packing your mod files	11
2.4.1 File packing method	11
2.4.2 Mod installation and uploading	13
<b>3. How to mod</b>	<b>15</b>
3.1 Hex mods	15
3.2 Other mods	18
<b>4. Conclusion</b>	<b>19</b>
<b>5. DRG Modding Discord</b>	<b>19</b>

# 1. Introduction and tools

## 1.1 The Basic Knowledge

[\(Back to Index\)](#)

There's still a lot of things to do in DRG even if you have all the cosmetics and promoted all characters to the max level. Welcome to DRG modding, you are going to like this game even more after learning how to mod it. If you find it too difficult just try the current mods that the modders have released and have some fun. I'm at 1k hours myself and I would have stopped playing if it wasn't for the amazing mods.

Some **VERY IMPORTANT THINGS** before starting:

1. Make sure you've **read the official info about DRG's mod support** in the links below. I'm not going to go into details here so make sure you read them to understand the basics.

This is the main post:

<https://steamcommunity.com/games/DeepRockGalactic/announcements/detail/2953787944888179529>

This is the FAQ with some additional info:

<https://www.deeprockgalactic.com/modding-support-faq>

And I will link 2 more when we get to the part that talks about uploading your mods.

2. **Mods are hosted here** <https://drg.mod.io/>. Use the modding menu inside the game to **create an account automatically so you can upload your mods**.
3. Your vanilla or modded save files are safe because mods will not corrupt/mess with it unless you use the tools maliciously. If you are concerned about this use the save clone and backup feature that the game provides in the menu.
4. **This guide only covers the bare minimum** so if you want to learn more make sure you check all the other guides in the #guides-and-tools channel in the [drg modding discord](#).

First you are gonna need some tools to start modding. You can get a .zip with the basic files in the discord. **I**You can always **use the blue hyperlinks through the guide and in the index** to quickly find what you need but **I highly recommend reading 1. and 2. from top to bottom if it's your first time.**

## 1.2 The Basic Tools

### **1. FSD-WindowsNoEditor.pak**

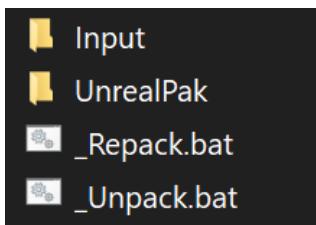
-> This is the **main file of the game**, which contains all the rest of the files. It's "paked" with Unreal Engine so we need to extract the files if we want to modify them.

-> You can find it in "...\\Program Files (x86)\\Steam\\steamapps\\common\\Deep Rock Galactic\\FSD\\Content\\Paks". You obviously need the game installed and **NEVER DELETE THIS FILE**, just copy it if you need it later. If you don't know how to access this folder go to Steam, right click DRG > Properties > Local Files > Browse Local Files. If you want to make a lot of mods it is recommended to **make a backup of the extracted files inside it** when we get to that point.

-> The **mods that you create and WANT TO TEST need to have a certain name to be loaded by the game** which should be the name that you want followed by "\_P". Mods uploaded to mod.io don't have this restriction so they don't need the "\_P", more about this in [2.4.2 Mod installation and uploading](#).

### **2. DRGPacker**

-> This tool will allow us to extract the files from [FSD-WindowsNoEditor.pak](#) and will also let us pack our own files. The **files that you extract will have the extensions .uexp and .uasset** which are Unreal Engine files. More on this on [2.1 Unpacking the game's files](#).



**For older users of the guide:** You don't need [UUE](#) anymore with the new version of the packer if you just want to make hex mods.

### **3. AssetEditor and UAssetGUI**

-> These are used to see the contents of the files and change values. You are going to **need at least 1 of them if you want to make mods that change a value/s** inside the game files. Both serve the same purposes and for basic use you can try both and see which one you like more, there will be an example in [3.1 Hex mods](#). The developer of one, UAssetGUI, is in our discord so ask him any questions. **If the tools don't work then download .NET Runtime: <https://dotnet.microsoft.com/download>**

#### 4. EmptyContentHierarchy

-> These are all the folders inside an extracted [FSD-WindowsNoEditor.pak](#) but empty, just the folders. This will be useful later to pack files and for UEE in other guides. **Each major update will have a different one**, read the .txt's inside the modding tools zip to learn how to get the latest version (you'll get them from a github or make them yourself with a Windows command).

_ModBPs	AI	Analytics	Art
Audio	Character	CharacterStructure	Cinematics
Collections	Critters	Developers	Enemies
Game	GameElements	Landscape	LevelElements
Localization	Maps	Movies	PersonalFolders
Rumble	StarterContent	TempContent	UI
WeaponsNTools			

#### 5. UModel

-> This tool will allow you to **extract the audio files, textures, 3d models...** This is different from [DRG Packer](#) because that one just extracts the .uexp and .uasset files but if you want to dive deeper and get the sounds, icons, 3d models... you are going to need this. You can find it here <https://www.gildor.org/en/projects/umodel>.

#### 6. FModel

-> This tool **will give you readable files from a pair of .uexp and .uasset files.** (**Readable means not in hexadecimal**). AssetEditor or UAssetGUI already parse the files but they may have problems and in my opinion it's easier to search for that value you want to change in FModel before you edit it.

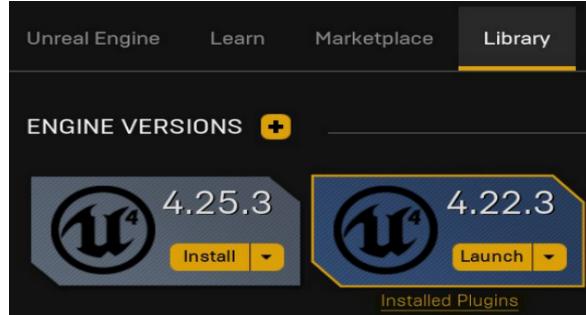
#### 7. Unreal Engine Editor (UEE)

-> If you want to make mods to **replace audio and textures** you need this. You can also **create your own Unreal Engine blueprints** (which are way more complex than hex editing mods or audio mods but can do more stuff).

-> You can find it here <https://www.unrealengine.com/en-US/>. Unfortunately you need to download the Epic Games Store even though we just want Unreal Engine and not the games but there's also a github with a quick google search that tells you how to download it without the store.

-> **VERY IMPORTANT:** You are going to **need the version that DRG is made with**, which is version 4.25. **This might change in the future.** For now, get the latest subversion so 4.25.X, X being whatever is the max in there. If you want **to create working 3d model replacement mods** you'll need the 4.25+ version (it's not the same) and need to get it from the official github.

Go to the Library, add a new version with the + symbol -> click the arrow next to install -> and the version. You **only need the basic installation** so you can configure it and remove everything but the basic stuff so the installation **takes only like 13 GB of storage instead of +30 GB.**



## 8. Other tools

[\(Back to Index\)](#)

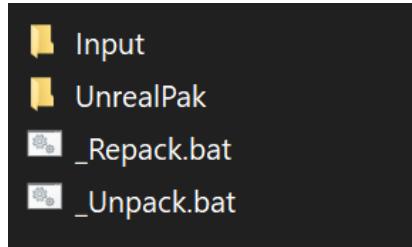
-> Those are the main tools you need but there's way more in the guides channel in [the modding discord](#) (tools for specific types of mods, console unlocker, research tools...).

## 2. Things to learn before modding

### 2.1 Introduction to DRG Packer

[\(Back to Index\)](#)

First you are going to need to **download the basic tools zip in our discord**. This is our fresh DRG Packer folder after extraction, it's pretty easy to use.

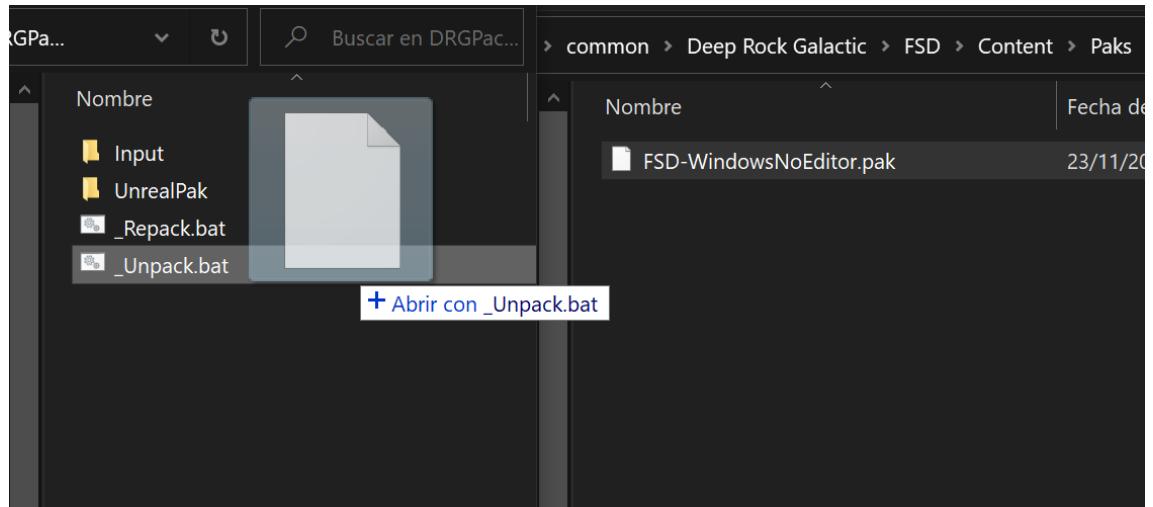


To pack files you will drag and drop the input folder into `_Repack`, to unpack the main FSD from the game you just drag and drop into `_Unpack`. Let's see a more detailed step by step guide now.

### 2.2 Unpacking the game's files

[\(Back to Index\)](#)

First of all **we want to extract all the files** from [FSD-WindowsNoEditor.pak](#). Start by going to your [DRG Packer](#) folder and your FSD folder (Check above if you forgot how to). Put them side by side and drag and drop [FSD-WindowsNoEditor.pak](#) into `_Unpack`.



Now wait for the extraction to complete. The command line window will say “Press any key to continue...” when it’s done you will see a new folder called “FSD-WindowsNoEditor”. The files of the game are inside it.

We are not going to use the contents of the “Engine” subfolder and the other folders inside “FSD” that are not “Content”. Just remove everything but the FSD->Content folder and you are set.

You should probably **make a backup of FSD with all the extracted files** and **don’t forget that you need to extract the files again for each update the game receives.**

## 2.3 Checking the content of the files

[\(Back to Index\)](#)

If you go to the “FSD” folder and then inside “Content” you will find the extracted folders. All of these contain the **pairs of .uexp and .uasset files** that we are going to need.

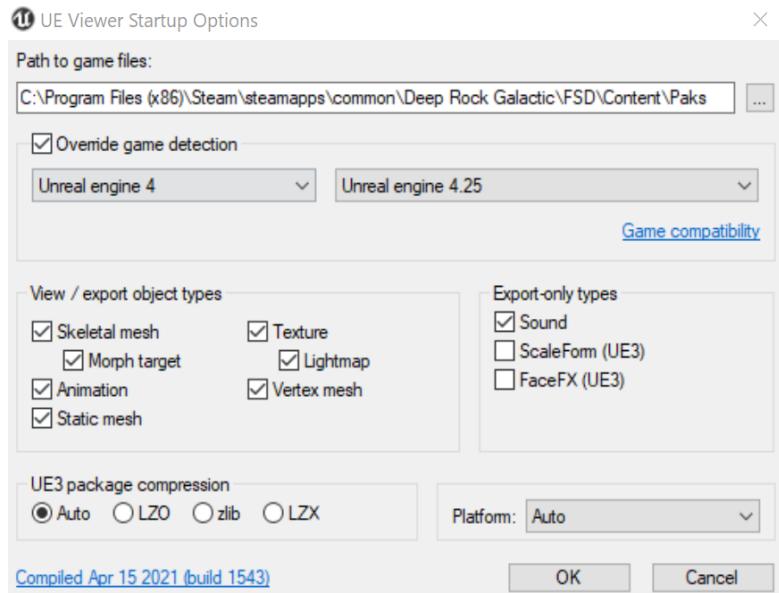


For example, if we go to “FSD\Content\WeaponsNTools\FlameThrower” we can see all the files for the flamethrower weapon. For editing values like upgrades or base stats we can open some of the .uexp files using a software like HxD to see the contents in hexadecimal or use [FModel](#) or the asset editors to have more readable text.

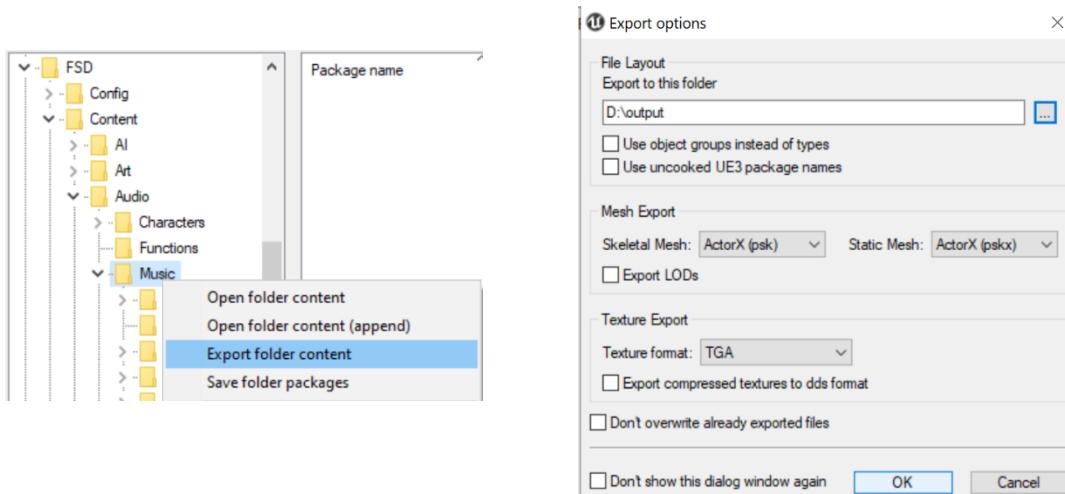
Offset(h)	00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	Decoded text
00000000	20 00 00 00 00 00 00 0F 00 00 00 00 00 00 00 00	.....
00000010	08 00 00 00 00 00 00 00 0D 00 00 00 00 00 00 00	.....
00000020	00 0E 00 00 00 00 00 00 00 05 00 00 00 00 00 00	.....
00000030	00 11 00 00 00 00 00 00 04 00 00 00 00 00 00 00	.....
00000040	00 00 00 00 FA 43 15 00 00 00 00 00 00 00 00 00 00	.....úC.....
00000050	00 00 00 00 00 49 00 00 00 00 00 00 00 00 00 08	.....I.....
00000060	00 00 00 00 01 00 00 00 00 21 00 00 31 45 33	.....!E3.....
00000070	41 43 43 30 30 34 39 43 32 45 36 46 32 32 44	ACC0049C2E6F22D
00000080	45 43 42 42 32 42 30 43 46 35 46 44 32 00 16 00	ECBB2B0CF5FD2...
00000090	00 00 48 69 67 68 20 50 72 65 73 73 75 72 65 20	..High Pressure
000000A0	45 6A 65 63 74 6F 72 00 0C 00 00 00 00 00 00 00	Ejector.....
000000B0	1D 00 00 00 00 00 00 00 74 00 00 00 00 00 00 00	.....t.....
000000C0	00 08 00 00 00 00 01 00 00 00 21 00 00 00 33	.....!..3
000000D0	45 31 38 42 31 31 33 34 43 36 37 30 44 36 34 37	E18B1134C670D647
000000E0	39 32 32 36 30 39 39 33 33 43 35 33 31 30 43 00	922609933C5310C.
000000F0	41 00 00 00 49 6E 63 72 65 61 73 65 73 20 74 68	A...Increases th
00000100	65 20 72 61 6E 67 65 20 6F 66 20 74 68 65 20 66	e range of the f
00000110	6C 61 6D 65 20 66 6F 72 20 6C 6F 6E 67 20 64 69	lame for long di
00000120	73 74 61 6E 63 65 20 69 6E 63 69 6E 65 72 61 74	stance incinerat
00000130	69 6F 6E 2E 00 0A 00 00 00 00 00 00 00 13 00 00	ion.....
00000140	00 00 00 00 04 00 00 00 00 00 00 00 00 00 01 00	.....
00000150	00 00 08 00 00 00 00 00 00 00 17 00 00 00 00 00	.....
00000160	00 00 04 00 00 00 00 00 00 00 FC FF FF 1A	.....FF0000

We are going to be working on this file later in [3.1 Hex mods](#) to learn how to change the values and search for them.

If you want to extract other things like sounds or textures you will need to use [UModel](#) and extract the desired file/s. For example, we open it to get one of Molly's sound files. You can select [FSD-WindowsNoEditor.pak](#) in your game folder or select the directory (my choice here) where you just extracted the .pak with [DRG Packer](#), and we select these settings and OK:



We navigate the folder tree until we find our file or a folder with all the files we need (you can extract the whole Audio folder for example if you want all the sounds in the game), and after selecting extract we select an output folder:



If it's a sound, 3d model, texture file... you will get a few files in the extraction folder.

## 2.4 Packing your mod files

[\(Back to Index\)](#)

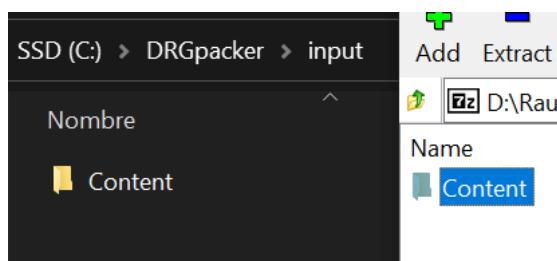
Using the same [DRG Packer](#) we used to unpack the main file, we are going to learn how to **pack the files that we changed/added into a new .pak file for the game to load.**

### 2.4.1 File packing method

I'm going to use the example from the flamethrower file before (imagine we already edited it even though that comes later in the tutorial). You should probably read about [Empty Content Hierarchy](#) if you haven't already.

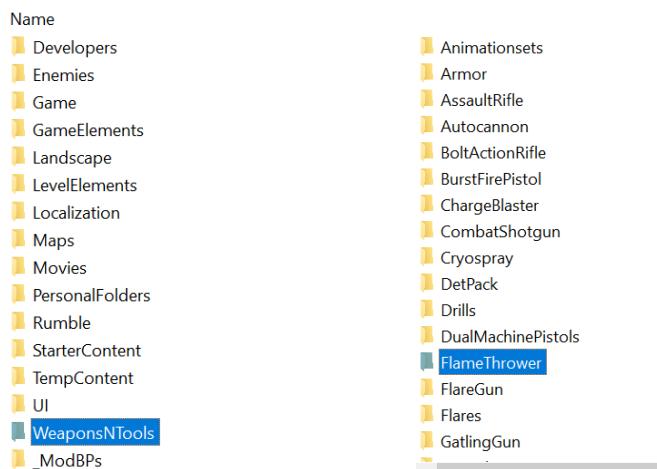
In the flamethrower example, the file is UPG\_FlameThrower\_A\_RangeIncrease which is inside FSD\Content\WeaponsNTools\FlameThrower. **To pack mods correctly we are going to need to replicate the same folder structure that we got on the output folder (unpacked [FSD-WindowsNoEditor.pak](#)) but inside the "Input" folder.**

You can just open the .zip from [Empty Content Hierarchy](#) (which has all the folders from the game but empty) and drop the "Content" folder in it inside "Input" and replace the placeholder one there.

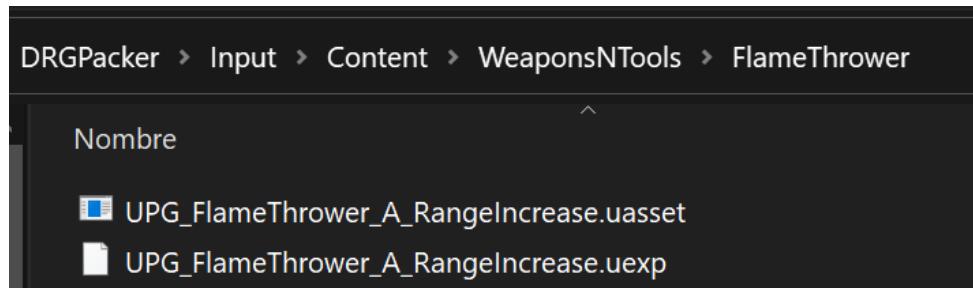


In this case we just want to work with 1 weapon so you don't need to have the rest of the folders. You can just drop the "WeaponsNTools" folder inside input -> Content.

You can also delete everything but the FlameThrower folder too. **I recommend having a backup of each folder structure for each mod you make**, for easier packing, instead of just having all of the empty folders and having to search where you put the files.



This will be the final path of the flamethrower files.



You can do the process before for **any number of files** that you modded.

**Be careful** when leaving folders in Input -> Content after packing because maybe you'll start making a new mod and forget you still had the old mod files in there.

Now just drag and drop the Input folder into \_Repack. (FSD here is just the extracted files from before, I recommend renaming them to the update where they are from).



Check the “**Added x files**” line. If you have for example 2 files in the mod check that the number is correct so it says “**Added 2 files**”. **If there's less than what you expect you forgot to put some files in the input folder, if it's more than expected then you forgot to remove files from a previously packed mod.**

```
D:\DRG_Modding\DRG_Mod_Tools\DRGPacker>.\UnrealPak\Engine\Binaries\Win64\UnrealPak.exe" "D:\DRG_Modding\DRG_Mod_Tools\DRGPacker\Input\_P.pak" -platform="Windows" -create="D:\DRG_Modding\DRG_Mod_Tools\DRGPacker\autogen.txt" -compress
LogPakFile: Display: Using command line for crypto configuration
LogPakFile: Display: Loading response file D:\DRG_Modding\DRG_Mod_Tools\DRGPacker\autogen.txt
LogPakFile: Display: Added 1 entries to add to pak file.
LogPakFile: Display: Collecting files to add to pak file...
LogPakFile: Display: Collected 2 files in 0.01s.
LogPakFile: Display: Creating pak D:\DRG_Modding\DRG_Mod_Tools\DRGPacker\Input\_P.pak.
LogPakFile: Display: DISABLING pak file index freezing (all must be true: HasPlatformInfo? true - TargetIs64Bit? true - NoRuntimeUnloading? true - Unencrypted? true, 'Windows' != 'Windows')
LogPakFile: Display: Added 2 files, 1665 bytes total, time 0.07s.
LogPakFile: Display: Compression summary: 53.56% of original size. Compressed Size 1008 bytes, Original Size 1882 bytes.

LogPakFile: Display: Used compression formats (in priority order) 'Zlib, '
LogPakFile: Display: Encryption - DISABLED
LogPakFile: Display: Unreal pak executed in 0.095168 seconds
Presione una tecla para continuar . . .
```

So, the **summary** of packing is that we only need to add or remove files inside the [EmptyContentHierarchy](#) (FSD and subfolders) with the correct folder structure compared to the vanilla files. It **can get pretty messy** if you forget to add/remove files when making new mods so (again) **I recommend having a backup of each folder structure for each mod you make**, for better organization and future updating.

## 2.4.2 Mod installation and uploading

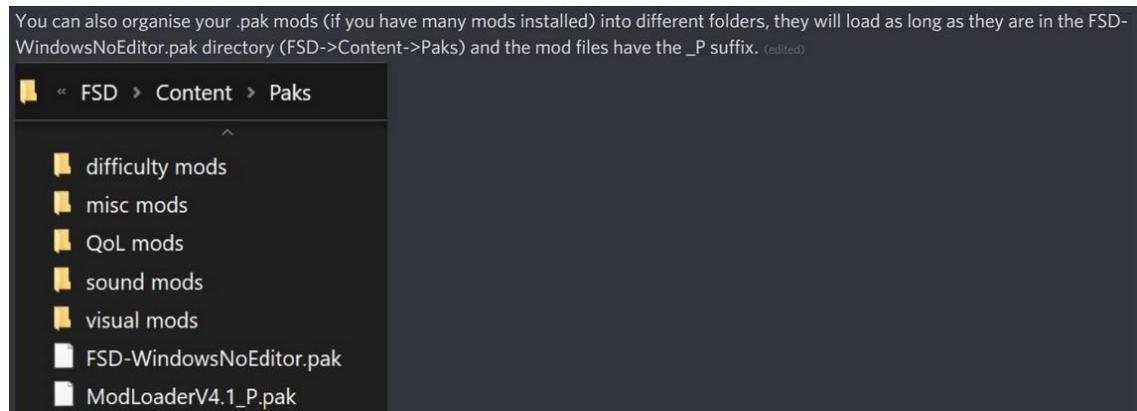
[\(Back to Index\)](#)

To install a mod FOR YOUR OWN TESTING PURPOSES (BEFORE UPLOADING), simply move the .pak than you made in the previous step to:

“...\\Steam\\steamapps\\common\\Deep Rock Galactic\\FSD\\Content\\Paks”

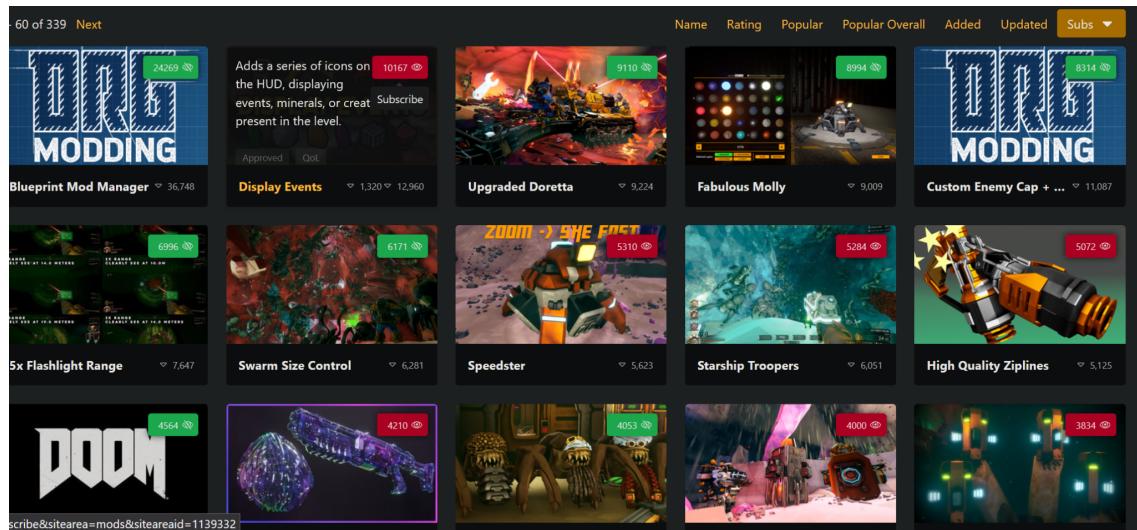
If you don't know how to open the game's folder follow this guide:

<https://steamcommunity.com/sharedfiles/filedetails/?id=760447682>



(Yes, installing the mod to test it is just moving your .pak in there. Unreal Engine automatically replaces the original file in [FSD-WindowsNoEditor.pak](#) when the game loads, with the file that you modified in your .pak).

To install a mod FOR PLAYING after it has been uploaded to <https://drg.mod.io/> you just browse mod.io or enter from inside the game in the modding menu, and click the big red eye button.



It will auto install and **don't forget to click APPLY CHANGES** after that at the bottom, the space rig will restart and your mod will be installed.

CATEGORY	NAME	AUTHOR	STATUS
PICKAXE INSTALL	PICKAXE BONKISFX VERSION 1.0	By BudmisterJesse	VERIFIED
APPROVED	5X FLASHLIGHT RANGE VERSION 3.46	By NatureBomCancer	APPROVED
APPROVED	SWARM SIZE CONTROL VERSION 1.1	By NatureBomCancer	APPROVED
APPROVED	UPGRADED DOOM	By putmekindasane	UPGRADED
VERIFIED	DOOM MUSIC VERSION 1.0	By Demolition	VERIFIED
VERIFIED	KEKW VERSION 1.00	By Demolition	VERIFIED

Compact View           

**TO UPLOAD A MOD (making it public)** after testing your *modname\_P.pak*, first remove the “\_P” and then read the following guidelines.

About mod categories:

<https://drg.mod.io/guides/mod-guidelines-and-status-categories>

And this other one to learn what you need to do before uploading a mod:

<https://drg.mod.io/guides/approval-process-and-checklist-for-upload>

After that make sure you have your description, name, category, thumbnail and .pak file ready and click on the + button on mod.io and follow the simple steps.

Games / Deep Rock Galactic      Rauliken

 **Deep Rock Galactic**  
This is the mod.io page for Deep Rock Galactic. Here you can find mods for the game made by others and also upload mods yourself. NB! This page is under development, guidelines, instructions, and more coming soon.  
[Learn more](#) and find the game on [Steam](#).

**Mods**     

     Date ad

## 3. How to mod

[\(Back to Index\)](#)

There's **different types of mods** that you can make for DRG. Hex mods (touch values and are easy to make if you know what you are looking for , there's also mods that change audio, textures... and blueprint mods to create Unreal Engine blueprint files with more complex behaviour.

In this guide I'll be explaining the basic hex mods. At the end I'll mention a few of the other guides to get you started with more complex mods. As always you can find everything #guides-and-tools in our modding discord

### 3.1 Hex mods

[\(Go to quick guide\)](#)

We have the hex mods, which **edit values inside the .uasset and .uexp files (all files are in pairs of these types)** and replace the original files with your new ones when the game loads. These values can be integers, text, booleans, floats... Most of the values should be easy to find just by using the armory inside DRG but you need to keep a few things in mind.

We **need to know the value that we are going to change but also the file where it is located**. You can check the [File Prefix List](#) made by @Elytnwaen.

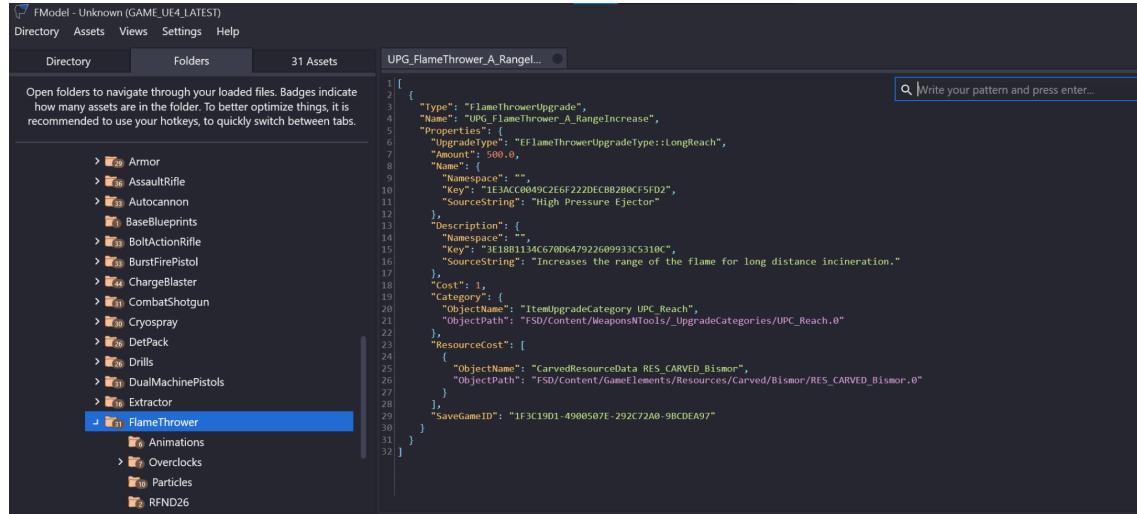
I'm gonna be working with the file that I used in the packing example before, in [2.4.1 File packing method](#). In this case our mod is going to have only one file which is called "UPG\_FlameThrower\_A\_RangeIncrease.uexp" and as you can tell by the prefix "UPG" this is the file that controls the range upgrade for driller's flamethrower.

I'm going to be using [UAssetGUI](#). If you want to learn more about how .uexp and .uasset files work download the DAUM guide made by @Drillboy Jenkins. Open the .uasset and explore a bit, to change a value go one of the categories:

Name	Type	Variant	Value	Value 2	Unit
0 UpgradeType	EnumProperty		EFlameThrowerUpgradeType	EFlame Thrower Upgrade Type	LongReach
1 Amount	FloatProperty		500		
2 Name	TextProperty	0	1E3ACCC049C26F222DECBB2B0CF5FD2	High Pressure Ejector	
3 Description	TextProperty	0	3E18B113AC670D647922609933C5310C	Increases the range of the flame for long distance incineration.	
4 Cost	IntProperty		1		
5 Category	ObjectProperty	4	UPC_Reach		
6 ResourceCost	ArrayProperty	ObjectProperty			
7 SaveGameID	StructProperty	Guid			
*					

Right there in the "Amount" field is the value we are looking for. It's 500 because the upgrade of +5 in game is measured in meters and it's centimeters in the files. Some values can be more obvious than this. We'll change it to 2000 so it becomes a 20 meter range upgrade.

If you can't find the value try using FModel. Open the main .pak from the game, navigate to the folder, go to the Assets tab on the top right and open the file. Use CTRL+F to search stuff. This is pretty useful in other files that have a lot of stuff but isn't really needed for this small change.



```

FModel - Unknown (GAME_UE4_LATEST)
Directory Assets Views Settings Help
Directory Folders 31 Assets UPG_FlameThrower_A_Range...
Open folders to navigate through your loaded files. Badges indicate how many assets are in the folder. To better optimize things, it is recommended to use your hotkeys, to quickly switch between tabs.
1 [
2   {
3     "Type": "FlameThrowerUpgrade",
4     "Name": "UPG_FlameThrower_A_RangeIncrease",
5     "Properties": {
6       "UpgradeType": "EFlameThrowerUpgradeType::LongReach",
7       "Amount": 500.0,
8       "Name": {
9         "Namespace": "",
10        "Key": "3E3AC0049C2EF220ECB8280CF5FD2",
11        "SourceString": "High Pressure Ejector"
12      },
13      "Description": {
14        "Namespace": "",
15        "Key": "3E1881134C6700647922609933C5310C",
16        "SourceString": "Increases the range of the flame for long distance incineration."
17      },
18      "Cost": 1,
19      "Category": {
20        "ObjectName": "ItemUpgradeCategory_UPC_Reach",
21        "ObjectPath": "FSD/Content/WeaponsNTools/_UpgradeCategories/UPC_Reach.0"
22      },
23      "ResourceCost": [
24        {
25          "ObjectName": "CarvedResourceData_RES_CARVED_Bismor",
26          "ObjectPath": "FSD/Content/GameElements/Resources/Carved/Bismor/RES_CARVED_Bismor.0"
27        }
28      ],
29      "SaveGameID": "1F3C19D1-4900507E-292C72A0-9BCDEA97"
30    }
31  ]
32 ]

```

Finally, we just save the file in UAssetGUI or AssetEditor with CTRL+S or File -> Save and we are ready to pack our mod. You will get 2 new files and the tools might make a backup (.bak) of the original ones.



You should know how to pack that pair of files (.uasset and .uexp) with one of the methods that I showed you earlier in [2.4 Packing your mod files](#). I used these exact files in both examples to make it easier to understand the whole process.

After packing, just follow [2.4.2 Mod installation](#) and you are ready to test your new flamethrower upgrade that will add +20 meters instead of +5. As you can see in the armory when I hover over the range upgrade it now gives us 20 extra range. You can try the range overclock too for some insane flame reach :P



Here's a quick guide of all the steps for HEX MODS:

1. Extract .pak files with [DRG Packer](#) by dragging the .pak into “unpack”.
2. (Optional) Read the files with [FModel](#) if you want more info about the value you want to change.
3. Open the .uexp file you want to change with a [UAssetGUI or AssetEditor](#) and edit the values you want.
4. Now to turn it back into .pak again follow these steps: (an example for the barrel kick force mod that modifies only PST\_BarrelKicking.uexp):
  - 4.1 Create the same folder structure inside DRGpacker -> input -> Content as the original extracted files had and add the modified files, so:  
...../GameElements/PawnStats/PST\_BarrelKicking.uexp
  - 4.2 Drag and drop the .txt to “pack” and check that there are no errors.
5. Put the generated new\_P.pak file in to test it:  
...\\Steam\\steamapps\\common\\DeepRockGalactic\\FSD\\Content\\Paks
6. After testing it to make sure it works, rename it to InsertNameHere.pak, you don't need the “\_P” anymore.
7. Upload to mod.io (Please read the guidelines, write proper descriptions and choose the correct category, you should have no issue if you read the info links)

And there's a few **important things about hex mods**:

1. If your game crashes, don't send the crash report to the devs. You probably changed the wrong value or changed file size without a proper tool like DAUM.
2. **Most but not all mods made this way are client side**, the meaning of this is that unless the host has the mod, or you are the host, it won't work. It's also like this for other mods like the difficulty mods (haz 6 etc) and blueprint mods. Audio mods are always client side.
3. As I explained in the packing tutorial, **you can have more than 1 file in a mod**. You **don't need to make 10 .pak files if you modify 10 files**, just make a single one by adding more files with the correct paths to the input->Content folder.
4. You **can change a lot of things with hex editing, not only the weapon's stats**. If for example you change a perk, you will need to **try the changes inside a mission** to make sure they work because those don't show up in the armory.
5. This type of modding is the most basic but It will complement the others. **For complex hex mods** like adding overclocks, changing character inventories... you could try doing it with the asset editors but I highly recommend using **DAUM**.

## **3.2 Other mods**

[\(Back to Index\)](#)

Now that you know the basics you are free to check all the other guides and tools.

For audio and blueprint mods the user @Buckminsterfullerene#6666 has made some great ones.

For model and texture replacement @Pacagma is your guy. Don't forget what I said earlier in the basic tools part about Unreal Engine 4.25+.

There's many more like the more complex hex mods with the DAUM guide, string replacement... Explore a bit in our discord and **use the search bar in #guides-and-tools**.

This marks the end of the basic guide but you should keep reading for some final tips and encouragement.

## **4. Conclusion**

[\(Back to Index\)](#)

I hope you found the guide simple enough to understand and that you make many great mods. It's been a long time, we started this community with 2-3 modders and a small private discord and now we have members in the thousands, the developers added official mod support and with this guide we have a new modder to welcome :)

If you ever feel frustrated when making mods don't forget that there's always someone willing to help, it's a very wholesome community. You can find a ton of info in the #mod-chat channel in our discord and there's even a channel where people stream themselves making mods so you can all learn together.

If you ever want to check some old mods for inspiration, extracting the files to see which ones they changed etc, this is the github where the old Update 33 mods were before the modding support update: <https://github.com/ArcticEcho/DRG-Mods>

See you in discord and in the mines, ROCK AND STONE!

## **5. DRG Modding Discord**

[\(Back to Index\)](#)

This is where we have our modding community. We post mods, previews, guides, chat about the game and have a good time.

Here's the invite link: <https://discord.com/invite/drg-modding>