

# Call of Duty 4

# Modding Tutorial:

This tutorial assumes that you have the Call of Duty 4 on PC, the Mod Tools ([http://www.infinityward.com/wiki/index.php?title=Mod\\_Tools\\_Download\\_and\\_Revision\\_History](http://www.infinityward.com/wiki/index.php?title=Mod_Tools_Download_and_Revision_History)) and at least a basic knowledge of computers.

This tutorial will cover the basic steps involved in making a mod based on *ModWarfare*. This tutorial will include basic coverage of weapon editing, skinning, scripting, custom sounds and effects editing (very basic coverage). It will do this by following the steps I went through making a mod from start to end.

## Fore Word:

Because this is a tutorial, I have been able to show you making a mod without making any errors. But I can tell you for sure that, just like every modder, you will make a lot of errors! It is normal to enter and exit the game every 20 secs trying to fix compile error.

There is also no single way to make a mod; this was just a single method. If I were making this mod for real, I would have been compiling it after every step to make sure no errors get past me.

Another point I will make is that although the mod in this tutorial does not make heavy use of the ModWarfare code, ModWarfare does contain features that you may want to use in another mod. For example, it gives users access to all weapons, and makes the settings more easily customizable.

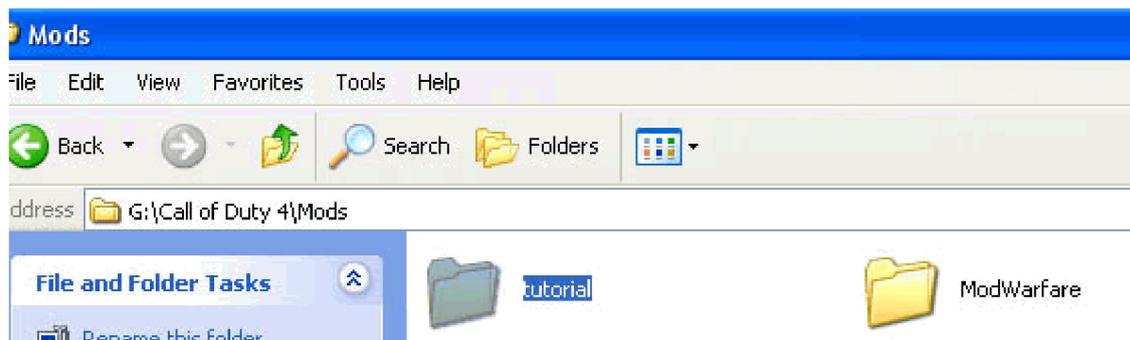
## **Tip:**

\*Run your mod in developer mode (/developer 1 in console) while you are developing it. This makes the game more sensitive to errors so you know if you are getting your code right.

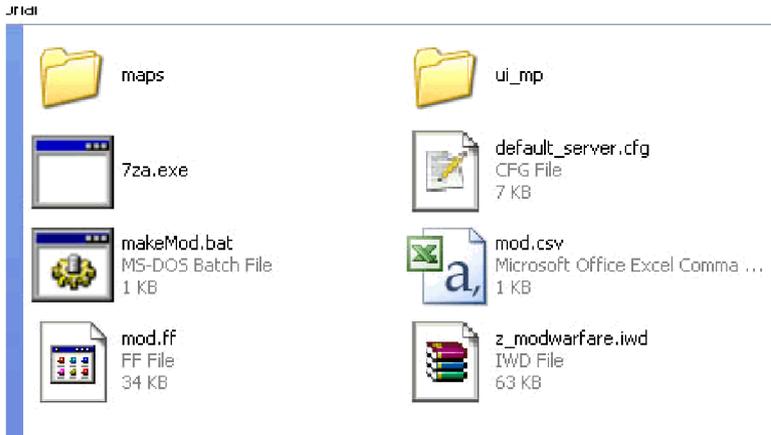
## **Preparation:**

Before we do anything we need to know what we want to be modding. The mod I will be making in this tutorial will turn the game into a rocket arena.

Now that I know what I want the mod to do, I can start preparing my work area. To do this I will need a folder in the CoD4/Mods to contain all of my files. In this case I will call it tutorial.



As you can also see, I have another folder called ModWarfare. This is a stock mod that comes with the game. I will be basing my mod on the code from ModWarfare. To do this I will need to copy the contents of the ModWarfare folder into my mod folder. My folder now looks like this:

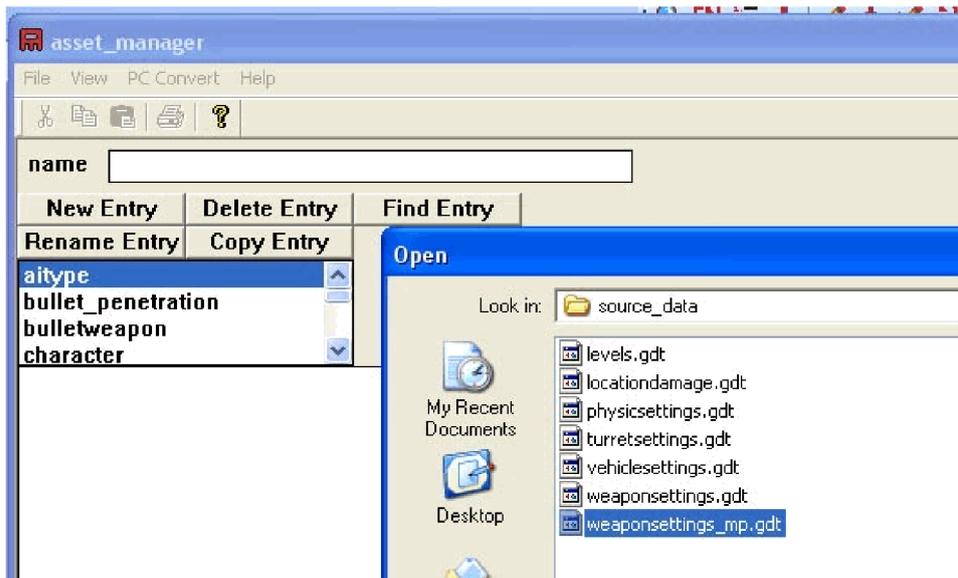


## **Getting Started:**

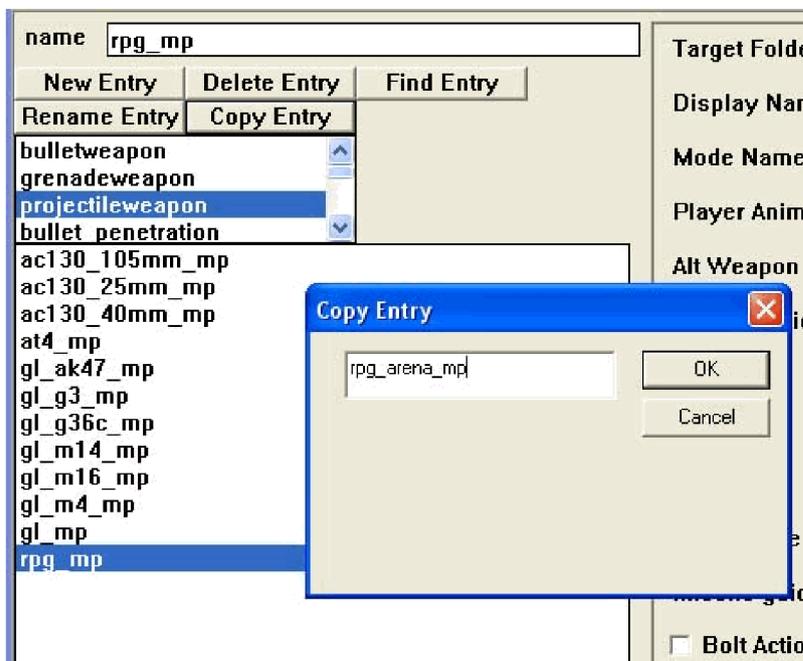
Now that we have the basic files that we will need we can get started on the modding. First I will be using the Asset Manager from the Mod Tools to edit the RPG to my liking. I know a lot of people are confused by the asset manager, so I will be using a lot of images for this process.

## **Asset Manager:**

First we need to open the asset manager. This is done through the CoD4 Compile Tools (<CoD4 directory>\bin\CoD4CompileTools) Run the exe and open asset manager from the Applications tab. Because I want to edit a multiplayer weapon, I click on File>Open>weaponsettings\_mp.gdt



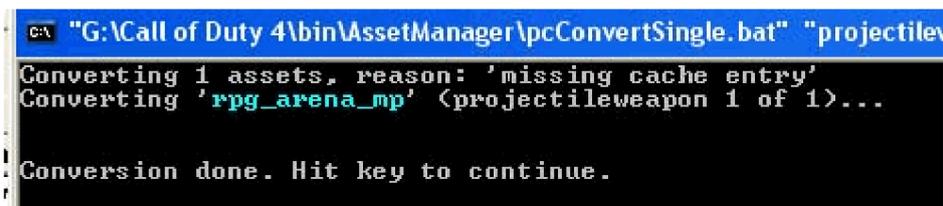
Once I have opened this file I select the “projectileweapon” category (because the RPG is classed as a projectile weapon) and find the entry for the RPG. Because I don’t want to edit the original weapon file (although it doesn’t make much difference) I select “Copy Entry” and enter a new name.



Now that I have a copy of the RPG I can start editing it to make a rocket arena mod. I will want to change several things relating to how the gun behaves. Because this would require a lot of screenshots, I will list the aspects I change (in order going down the asset manager window):

- \*Inventory type – The RPG is an item by default, I want it to be a primary weapon.
- \*Uncheck ADS (aim down sight) fire only.
- \*Max ammo and start ammo changed to 1000 – an arbitrary large number.
- \*Destabilize Distance – changed to 1000
- \*Reload Time – changed to 2
- \*Hip Spread settings – just generally decreased the spread
- \*Ammo Counter clip style – changed to none (people don't need to know how much ammo they have)

Now that I have edited the weapon, it is time to convert it for use in-game. Do this by pressing F10 (or navigate through the PC Convert menu). When converting is done, you should have a screen that looks like this:



```
C:\ "G:\Call of Duty 4\bin\AssetManager\pcConvertSingle.bat" "projectileweap
Converting 1 assets, reason: 'missing cache entry'
Converting 'rpg_arena_mp' (projectileweapon 1 of 1)...
Conversion done. Hit key to continue.
```

This will create the weapon file in a folder called <CoD4>\raw\weapons\mp\  
Create some new folders in the mod folder called weapons\mp\ and copy the newly created weapon file into that folder.

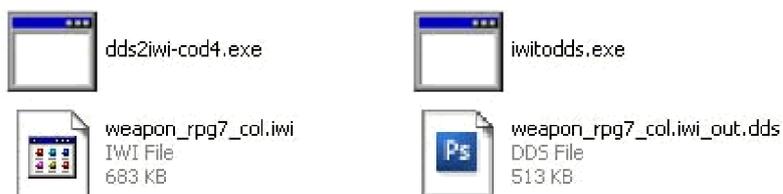


Later on in the process of making this mod I will be coming back to asset manager when I want to add some of the things I have made.

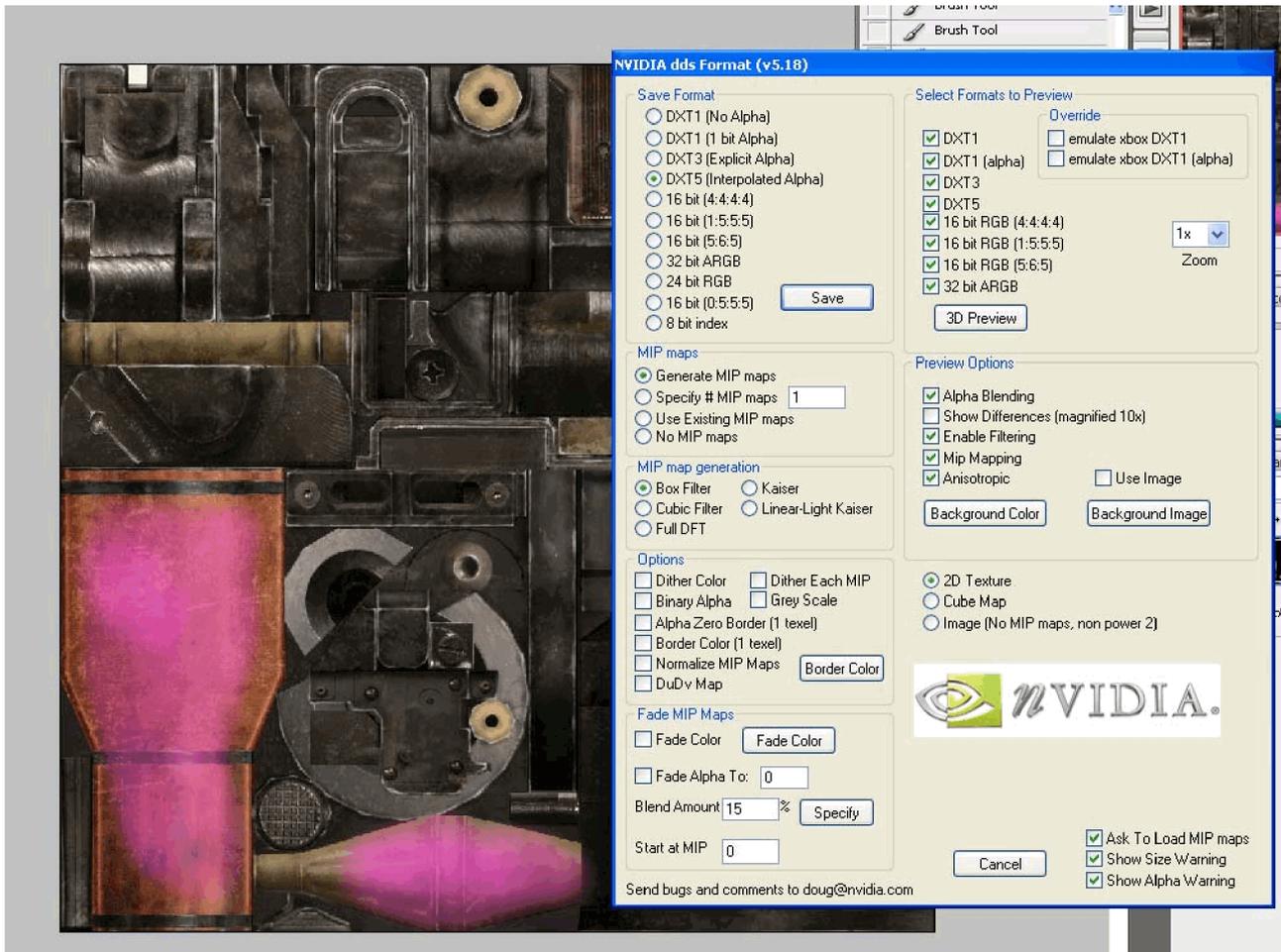
### **Skinning:**

Now that I have a weapon to use, I might want to make it look a bit more special than usual. I will need to do a bit of skinning. In order to do this I need to get the image for the RPG skin. The weapon skins are usually found in iw\_04.iwd in the “main” folder. These can be extracted with any archiving software. The file we are after is called “weapon\_rpg7\_col.iwi.” You can tell the skin images from the rest because they contain \_col or \_c in the name.

I have extracted the image file to an arbitrary folder so that I can edit it. Iwi files need to be converted to \*.dds using the iwitodds.exe provided with this tutorial (I didn't write the programs provided with this tutorial, full credit goes to their authors). To work these programs just drag and drop the iwis onto the exe. After this step I have a folder that looks like this:



The now I can open the dds file in photoshop using the dds plugin. There will be a message asking if you want to show existing mip maps, select no. Unfortunately I have the artistic talent of a retarded baboon, so I will keep the editing simple. After I have finished editing the skin I save it as a dds file using the options shown:



Beautiful isn't it ☺

I then use the `ddstoiwi-cod4.exe` program to convert the dds back to an iwi. And place it in a folder called `images` in my mod folder and rename it back to `weapon_rpg7_col.iwi`.



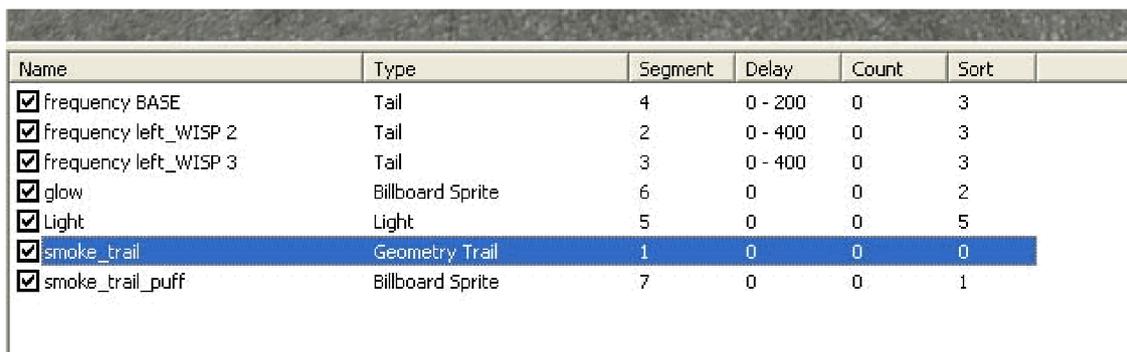
I will deal with adding it to the game a bit later when I bring the whole mod together.

## Editing Effects:

This, and menu scripting, are my most troublesome areas. So I will keep it quick and painless. I will start by launching the EffectsEd program from the CoD4 Compile Tools I mentioned earlier. This program always takes awhile to load (for me at least). You may have noticed (if you were following along at home) that in the asset manager it says the effect for the trail of the RPG rockets is called “smoke\_geotrail\_rpg.efx” This can be found in <CoD4>\raw\fx\smoke\

For the purposes of this tutorial I am only going to change the colour of the smoke, because I think pink is so much cooler than grey/white. The best way to learn how to use EffectsEd is to open different effects and, in another EffectsEd window, copy the settings for the different effects to replicate the effect and just experiment.

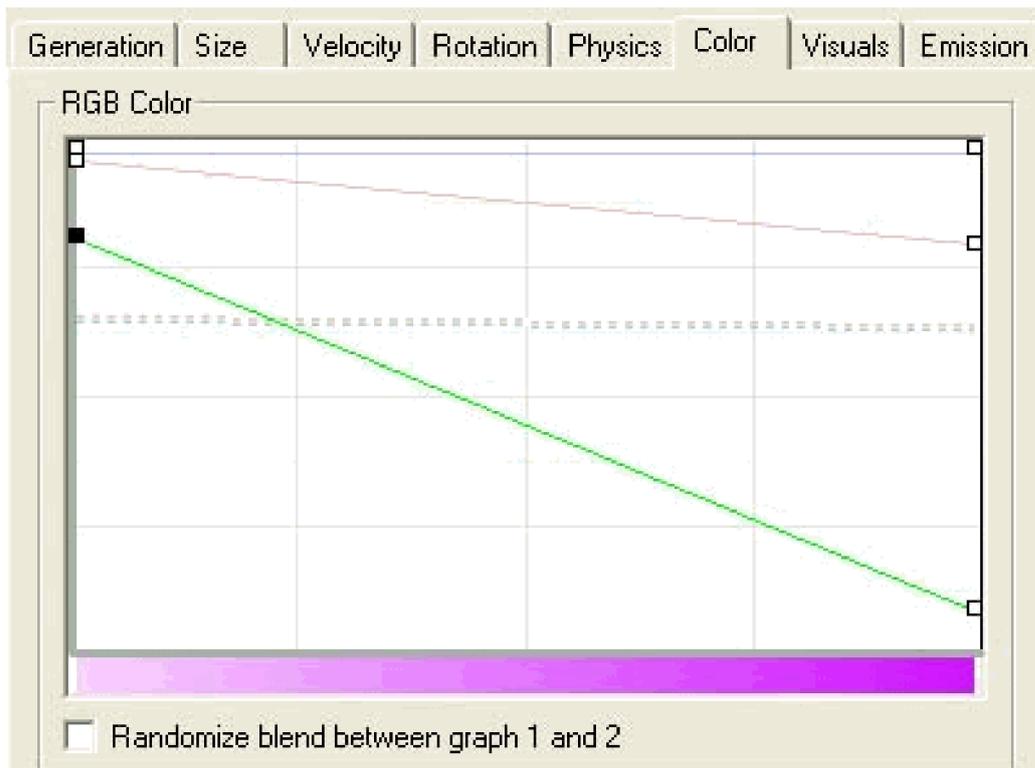
But for now I am only interested in the smoke colour. I will begin by selecting the part of the effect I want to edit.



Name	Type	Segment	Delay	Count	Sort
<input checked="" type="checkbox"/> frequency BASE	Tail	4	0 - 200	0	3
<input checked="" type="checkbox"/> frequency left_WISP 2	Tail	2	0 - 400	0	3
<input checked="" type="checkbox"/> frequency left_WISP 3	Tail	3	0 - 400	0	3
<input checked="" type="checkbox"/> glow	Billboard Sprite	6	0	0	2
<input checked="" type="checkbox"/> Light	Light	5	0	0	5
<input checked="" type="checkbox"/> smoke_trail	Geometry Trail	1	0	0	0
<input checked="" type="checkbox"/> smoke_trail_puff	Billboard Sprite	7	0	0	1

(The display of the different effect elements can be disabled by unchecking the check boxes)

Next I will navigate to the colour tab. Here I will change the colour of the various effects.



You may notice the colour gradient from light to dark pink. This represents the change in colour with respect to time. I.e. the smoke will change from light to dark pink over its lifetime. I will repeat this with the other elements of the effect that I want to change.

Now that I have my brand new colourful effect I will save it in the fx folder with a name different to the original (I don't want to overwrite the original). I will also copy this to a folder called fx in my mod folder.



Once this is complete we must tell the weapon to use this trail. To do this we need to open up Asset Manager again and find the attribute for the RPG called “Trail Effect” and point it to our new effect. Then we must reconvert it, and replace the copy we already have in the mod folder.

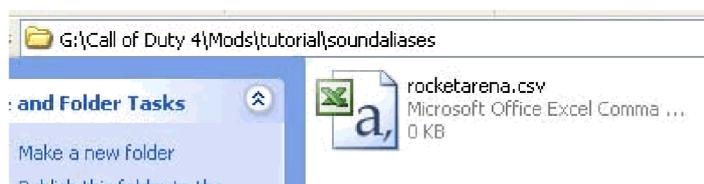


### Custom Sounds:

To add custom sounds to your mod you will need to create soundaliases for the sounds. I will be adding a custom sound for the RPG firing. I found the sound I want to use; now I need to setup a soundalias for this sound.



To start, a soundaliases folder needs to be made in the mod folder. In this folder I created a text file and renamed it to rocketarena.csv.



The current soundaliases for the RPG is inside the <CoD4>\raw\soundaliases\common.csv file. I open this file and search for the current sound aliases, copy them, and paste them in my file in my mod folder. I then edit the name of the sound file to the one I have chosen:

```
1 name,sequence,file,vol_min,vol_max,vol_mod,pitch_min,pitch_max,dist_min,dist_max,volumefalloffcurve,startdelay,speakermap,reverb,lfe_percentage,center_percent
2
3 weap_rpg_fire_plr,,rocket.wav,0.94,0.99,wpnplyr,0.75,0.85,500000,,local,,,
4 weap_rpg_fire_npc,,rocket.wav,0.79,0.84,wpnai,0.85,0.95,7,10920,weapon,,,,
```

The first line is between the start of the aliases and the end of the junk at the start. It is required for the compile to work. You can get it on page 18.

Another point to know is that while wav files can be stored in the mod.ff, mp3 files need to also be in the iwd.

**Scripting:** I am a java programmer so if you use different terminology, tough 😊

The purpose of this tutorial is to show you (the reader) the process I undergo to make a mod, not to teach you how to script. In light of that, do not expect a step by step walkthrough on how to script. The best place for scripting information is the Infinity Ward Scripting index (<http://www.infinityward.com/Script/index.htm>). CoD4 uses a language called QuakeC. Google can provide extra information about QuakeC.

This mod will not use very advanced scripting, but will show some basic scripting and where you are able to put your code. The methods shown here are not the be all and end all. Code however you are comfortable with. All of my scripting is available with this tutorial.

To start with I will make a new folder to contain the scripts I am going to write. In this case I will call it “rocketarena.” Due to the nature of scripting, I will comment the code instead of explaining all of the different parts in here. Gsc files are just plain text (\*.txt) files that have been renamed.

I will however discuss a few methods of loading your scripts, and other methods of modding. In this example I have used a call to my scripts in the main() method of the file called \_load.gsc. This file is executed when a new map is loaded. There are a few other ways of achieving similar results. For example; you can edit the map gsc files, however this will result in your mod not working in custom maps; you can also edit/create new gametypes, this means that you will need to specifically tell the game to load that gametype.

On top of adding your own scripts, it is important to know about other useful script files. For example, \_globallogic.gsc contains methods that are called in the event of player damage and death; \_weapons.gsc contains methods that monitor weapons such as nades, claymores, c4, etc.

## **Compiling the Mod:**

### **Preparing the compiler:**

Now that I have made all of the changes I want to make, it is time to prepare the tools to compile the mod.

First I will edit MakeMod.bat. I open this file using a text editor and change any references to ModWarfare to my own mod – in this case it is called “tutorial.” Next I will need to add lines that will tell it to add the extra files I have created the mod.

```
1 del z_tutorial.iwd
2 del mod.ff
3
4 xcopy ui_mp ../../raw/ui_mp /SY
5
6 xcopy fx ../../raw/fx\ /SY
7 xcopy maps ../../raw/maps\ /SY
8 xcopy weapons ../../raw/weapons\ /SY
9 xcopy soundaliases ../../raw/soundaliases /SY
10 xcopy rocketarena ../../raw/rocketarena /SY
11 xcopy sound ../../raw/sound /SY
12
13 copy /Y mod.csv ../../zone_source
14 cd ../../bin
15 linker_pc.exe -language english -compress -cleanup mod
16 cd ../../mods/tutorial
17 copy ../../zone/english/mod.ff
18
19 7za a -r -tzip z_tutorial.iwd images
20 7za a -r -tzip z_tutorial.iwd weapons
21
22 pause
```

In this file, lines 6-11 tell the compiler to copy the folder from your mod folder to the raw folder, and /SY means to copy subdirectories and suppress prompts to overwrite files. Lines 19 and 20 tell the compiler to make an iwd with the name z\_tutorial.iwd containing the folders images and weapons. These lines should be edited to contain what you have in your mod folder.

After this step, I will also need to edit the mod.csv file. This tells the compiler what files to store in the mod.ff.

The first word is the type of file, then after that is the path to the file. As you can see I have the line “weapon,mp/rpg\_arena\_mp” in my mod.csv. When dealing with weapons that are normally in the game, you don’t need to add the line for the weapons. But because the weapon has a new name, I need to include it in the mod.csv.

Tutorial by AintNoMeInTeam  
Xfire: aintnomeinteam

```
menufile,ui_mp/scriptmenus/changeclass_marines_mw.menu
menufile,ui_mp/scriptmenus/changeclass_opfor_mw.menu
menufile,ui_mp/scriptmenus/changeclass_mw.menu
fx,smoke_rpg_trail_pink
weapon,mp/rpg_arena_mp
image,weapon_rpg7_col
rawfile,maps/mp/gametypes/_class.gsc
rawfile,maps/mp/gametypes/_globallogic.gsc
rawfile,maps/mp/gametypes/_hardpoints.gsc
rawfile,maps/mp/gametypes/_healthoverlay.gsc
rawfile,maps/mp/gametypes/_menus.gsc
rawfile,maps/mp/gametypes/_missions.gsc
rawfile,maps/mp/gametypes/_modwarfare.gsc
rawfile,maps/mp/gametypes/_rank.gsc
rawfile,maps/mp/gametypes/_teams.gsc
rawfile,maps/mp/gametypes/_weapons.gsc
rawfile,maps/mp/_load.gsc
rawfile,rocketarena/setup.gsc
rawfile,rocketarena/rocketarena.gsc
rawfile,rocketarena/weapons.gsc
sound,rocketarena,,all_mp
```

## The Compile:

Phew! If you've stuck with me for this long, congratulations!  
Its time for the compile!!

To compile your mod, simple doubleclick on MakeMod.bat!

Here is the output for me:

```
G:\Call of Duty 4\Mods\tutorial>del z_tutorial.iwd
```

```
G:\Call of Duty 4\Mods\tutorial>del mod.ff
```

```
G:\Call of Duty 4\Mods\tutorial>xcopy ui_mp ..\..\raw\ui_mp /SY
ui_mp\cac_ingame.inc
ui_mp\cac_loadout_ingame.inc
ui_mp\scriptmenus\change_class_marines_mw.menu
ui_mp\scriptmenus\change_class_mw.menu
ui_mp\scriptmenus\change_class_opfor_mw.menu
5 File(s) copied
```

```
G:\Call of Duty 4\Mods\tutorial>xcopy fx ..\..\raw\fx\ /SY
fx\smoke_rpg_trail_pink.efx
1 File(s) copied
```

```
G:\Call of Duty 4\Mods\tutorial>xcopy maps ..\..\raw\maps\ /SY
maps\mp\_load.gsc
maps\mp\gametypes\_class.gsc
maps\mp\gametypes\_globallogic.gsc
maps\mp\gametypes\_hardpoints.gsc
maps\mp\gametypes\_healthoverlay.gsc
maps\mp\gametypes\_menus.gsc
maps\mp\gametypes\_missions.gsc
maps\mp\gametypes\_modwarfare.gsc
maps\mp\gametypes\_rank.gsc
maps\mp\gametypes\_teams.gsc
maps\mp\gametypes\_weapons.gsc
11 File(s) copied
```

```
G:\Call of Duty 4\Mods\tutorial>xcopy weapons ..\..\raw\weapons\ /SY
weapons\mp\rpg_arena_mp
```

1 File(s) copied

```
G:\Call of Duty 4\Mods\tutorial>xcopy soundaliases ..\..\raw\soundaliases /SY  
soundaliases\rocketarena.csv
```

1 File(s) copied

```
G:\Call of Duty 4\Mods\tutorial>xcopy rocketarena ..\..\raw\rocketarena /SY  
rocketarena\rocketarena.gsc  
rocketarena\setup.gsc  
rocketarena\weapons.gsc
```

3 File(s) copied

```
G:\Call of Duty 4\Mods\tutorial>xcopy sound ..\..\raw\sound /SY  
sound\rocket.wav
```

1 File(s) copied

```
G:\Call of Duty 4\Mods\tutorial>copy /Y mod.csv ..\..\zone_source  
1 file(s) copied.
```

```
G:\Call of Duty 4\Mods\tutorial>cd ..\..\bin
```

```
G:\Call of Duty 4\bin>linker_pc.exe -language english -compress -cleanup mod  
Fastfile 1 of 1, "mod": [ver. 5] process...link...compress...save...done.
```

```
G:\Call of Duty 4\bin>cd ..\mods\tutorial
```

```
G:\Call of Duty 4\Mods\tutorial>copy ..\..\zone\english\mod.ff  
1 file(s) copied.
```

```
G:\Call of Duty 4\Mods\tutorial>7za a -r -tzip z_tutorial.iwd images
```

```
7-Zip (A) 4.42 Copyright (c) 1999-2006 Igor Pavlov 2006-05-14  
Scanning
```

```
Creating archive z_tutorial.iwd
```

```
Compressing images  
Compressing images\weapon_rpg7_col.iwi
```

```
Everything is Ok
```

```
G:\Call of Duty 4\Mods\tutorial>7za a -r -tzip z_tutorial.iwd weapons
```

```
7-Zip (A) 4.42 Copyright (c) 1999-2006 Igor Pavlov 2006-05-14
```

```
Scanning
```

Tutorial by AintNoMeInTeam  
Xfire: aintnomeinteam

Updating archive z\_tutorial.iwd

Compressing weapons  
Compressing weapons\mp  
Compressing weapons\mp\rpg\_arena\_mp

Everything is Ok

G:\Call of Duty 4\Mods\tutorial>pause  
Press any key to continue . . .

If your output is like this when you compile, it is time to harvest  
the fruits of your labour 😊



**Notes:**

\*Thanks to IW for making CoD4 and the Mod Tools

\*Thanks to the people who made the iwitodds/ddstoiwi tools and DDS plugin for photoshop (I can't find your names); I get no credit for making those tools.

**Sound Alias Line:**

name,sequence,file,vol\_min,vol\_max,vol\_mod,pitch\_min,pitch\_max,dist\_min,dist\_max,channel,type,probability,loop,masterslave,lo  
adspec,subtitle,compression,secondaryaliasname,volume falloff curve,startdelay,speakermap,verb,lfe percentage,center  
percentage,platform,envelop\_min,envelop\_max,envelop  
percentage,conversion