REDmod

REDmod is a set of command line tools for modding *Cyberpunk 2077* that allow you to compile scripts and tweak source files, stage mods for use in-game and also to extract files from the game, export certain assets from it into modifiable formats and import them back into the form used by the game.

The main hub of the REDmod is redmod.exe, which executes different operations depending on the arguments provided. The easiest way to use REDmod is through the Windows command prompt or batch files.
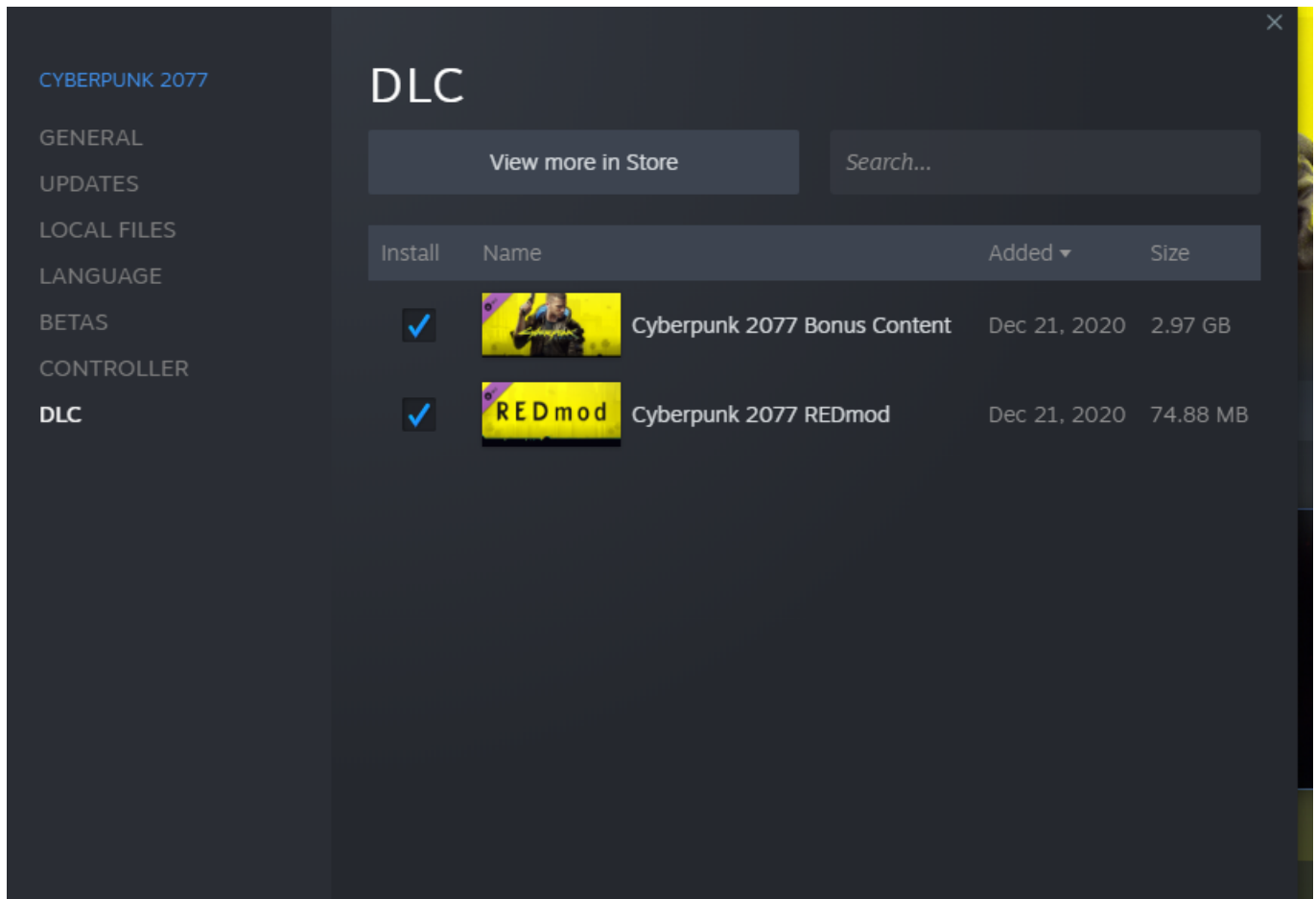
Installation

> To start creating mods or using mods for *Cyberpunk 2077* you must first install REDmod.

REDmod is shipped as free DLC with the game. You need to add it inside the platform store (GOG, Steam, Epic).

When REDmod is installed correctly you will find a new folder `tools/redmod` inside your game root directory, with subfolders: `bin`, `scripts`, `tweaks`.
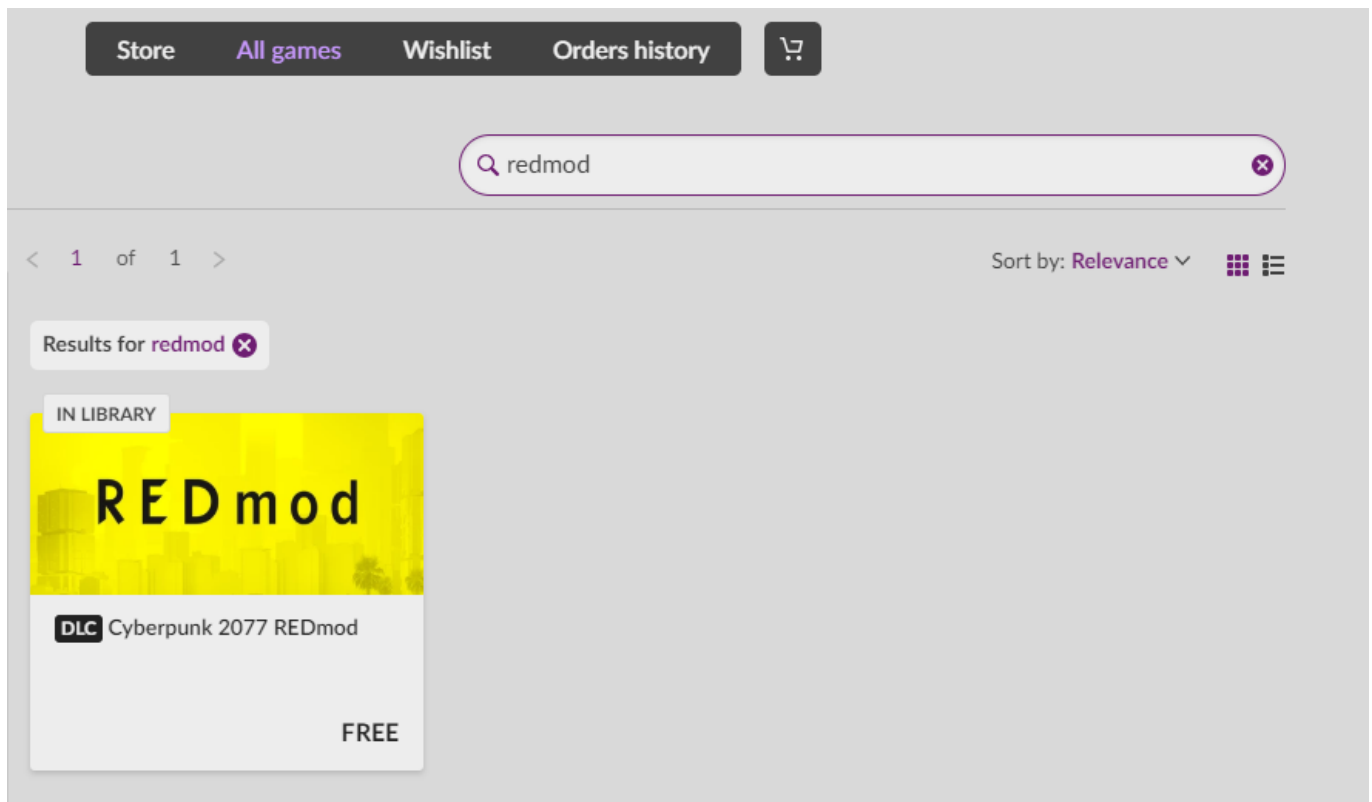
Usage

REDmod is a tool to stage installed mods to work with a modded game.

There are also various commands for modders to work with the game's assets.

Using Mods

The game loads mods from this path: `<Cyberpunk 2077>/mods/<name>`.

> EXAMPLE: A simple archive mod `C:\Gog\Cyberpunk 2077\mods\modA\archives\mod_a.archive`.

You need to **deploy** your installed mods before playing a modded game.

You can handle deployment and load order with

- the **REDlauncher** that comes with the game
- **Vortex** Mod Manager from NexusMods
- or use the REDmod WolvenKit integration and manage your mods easily from within the **WolvenKit Mod Manager**.

You need to **start a modded game** in order to load the deployed mods in game.

> Legacy archive mods in `<Cyberpunk 2077>/archive/pc/mod` are still supported but their load order cannot be managed with REDmod.

Deploying Mods and Starting the Game

REDmod can stage installed mods to work when the game starts with the `modded` flag:

- scripts in `<Cyberpunk 2077>/mods/<name>/scripts` are compiled into a modded script blob
- tweak files in `<Cyberpunk 2077>/mods/<name>/tweaks` are compiled into a modded tweakDB blob
- sound files in `<Cyberpunk 2077>/mods/<name>/customSounds` are staged to load

In order to utilize deployed mods with REDmod, you need to run the game in a specific *modded* mode. You can do this by launching the game through WolvenKit or manually by specifying a command parameter.

Deploying Mods - REDlauncher
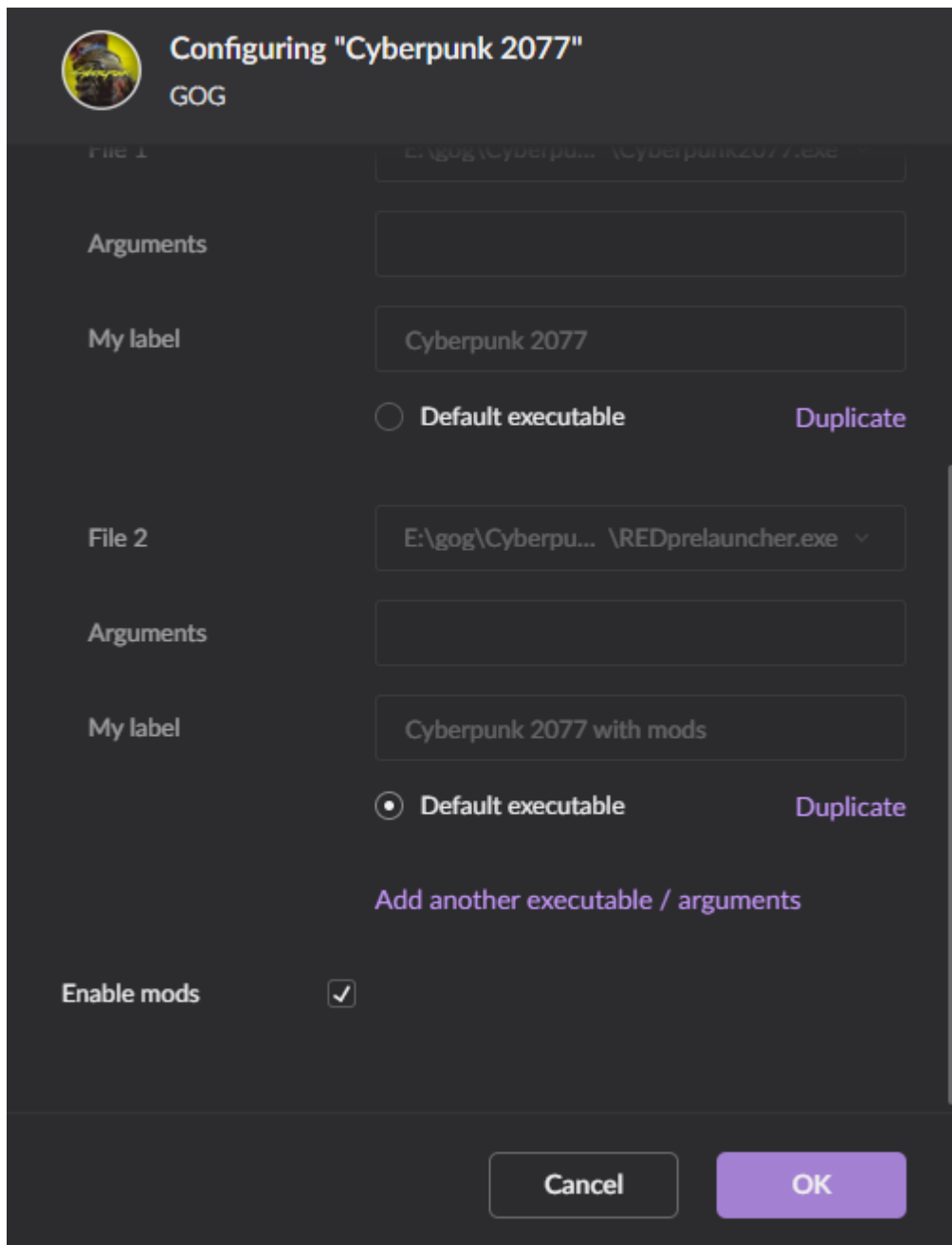
GOG REDprelauncher

- Open the `Cyberpunk 2077/REDprelauncher.exe`
- Click the **Settings** button and **Enable mods**
- Click **Play**
- A window will open showing progress on mod deployment.
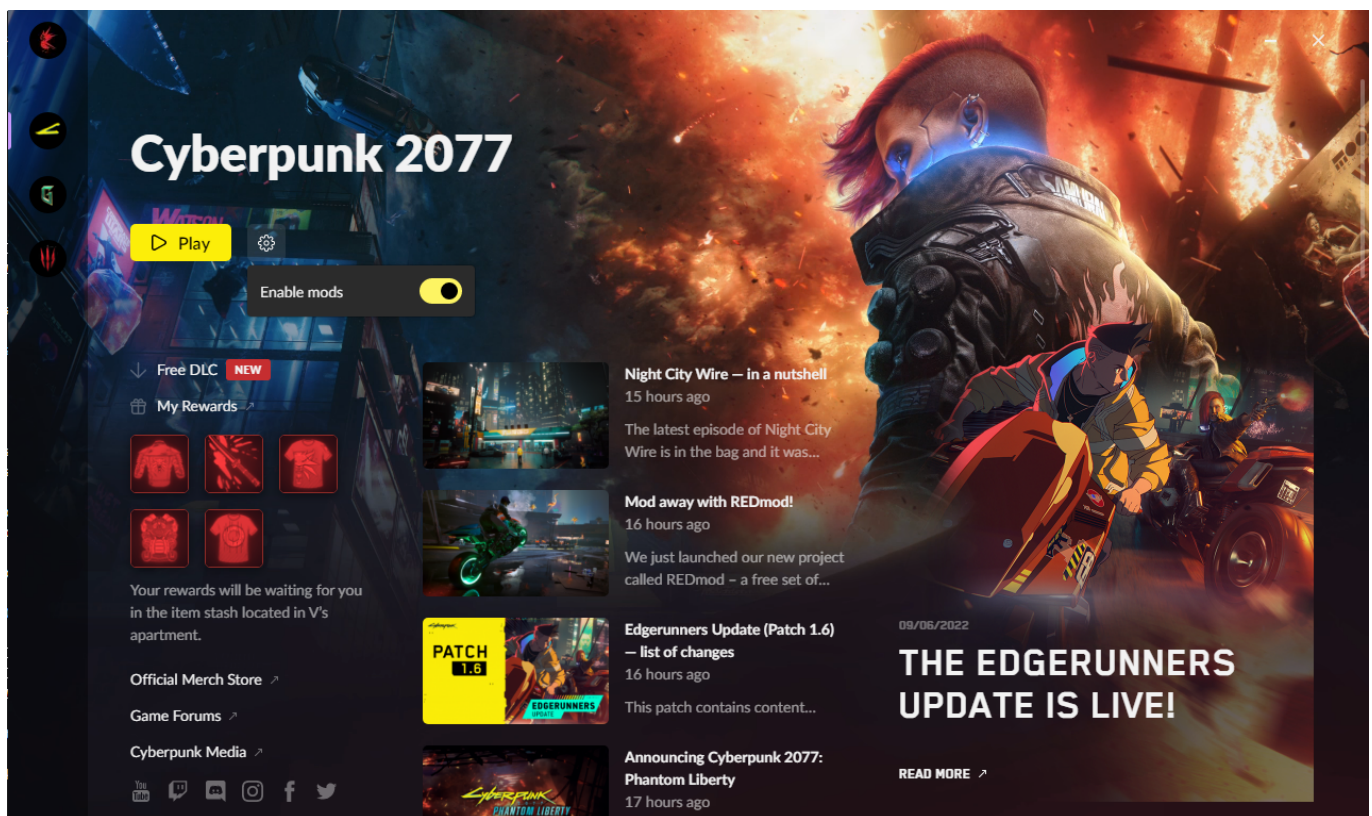
GOG Galaxy

- Open GOG - *Cyberpunk 2077*
- Click **Manage Installation** -> **Configure ...**
- Open the **Features** Tab
- Select **File 2** `REDprelauncher.exe` as default executable
- Select **Enable mods**
- Click **OK**
- Play the game

- A window will open showing progress on mod deployment.

Steam and Epic

- Navigate to *Cyberpunk 2077*
- Click **Play**
- Click the **Settings** button and **Enable mods**
- Click **Play**
- A window will open showing progress on mod deployment.

Conflicts and Load Order

Conflicts are handled on a per-file basis, but can be handled by specifying a specific *load order*.

Example: modA contains a script file `C:\Gog\Cyberpunk 2077\mods\modA\scripts\cyberpunk\player\player.script`. **modB** contains a script file with the same name `C:\Gog\Cyberpunk 2077\mods\modB\scripts\cyberpunk\player\player.script`.

This will discard all changes **modB** made, since by default, REDmod and the game loads mods alphabetically (modA overrides changes in modB).

However, you can specify a load order with REDmod, which the game will respect.

Advanced Topics

Deploying Mods - Manually

The REDmod `deploy` command stages installed mods to work when the game starts with the `modded` flag.

Deploying Mods - WolvenKit Integration

- Open the WolvenKit Mod Manager.
- Click the *Deploy Mods* button in the toolbar.

You will see all installed mods in your game. Enable the mods you would like to play with.

Handling Conflicts - Manually

You can specify a specific load order with the **-mod** parameter:

```
redmod deploy -root="C:\Gog\Cyberpunk 2077" -mod=modB,modA,modC
```

This will override changes to `\cyberpunk\player\player.script` that **modA** makes in favor of **modB**. It will also tell the game to load **modB** before **modA** and **modC**.

Handling Conflicts - WolvenKit integration

- Open the WolvenKit Mod Manager.
- Adjust the load order by dragging and dropping rows.
- Click the *Deploy Mods* button.

You can adjust the load order by dragging and dropping the installed mods inside the mod manager.

When you click the *Deploy Mods* button, the current load order will be forwarded to redmod.exe and mods will be deployed and loaded in this order.

Starting a Modded Game - Manually

To start a modded game that loads mods deployed by REDmod use the `-modded` flag on **Cyberpunk2077.exe**

Example:

```
Cyberpunk2077.exe -modded
```

> INFO The executable is located at `<Cyberpunk 2077>\bin\x64\Cyberpunk2077.exe`

Modding Guide

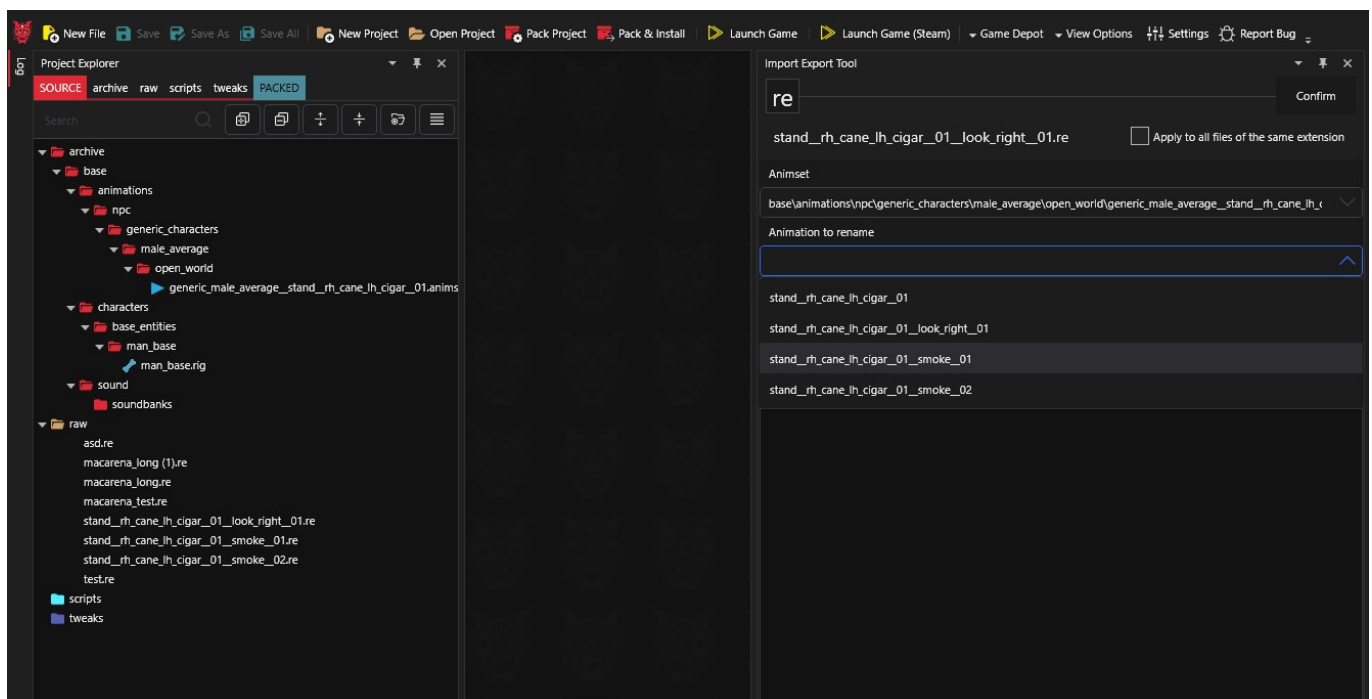## Animation Modding

Animation Modding - WolvenKit Integration

The REDmod animation import tool can be used as a Plugin in WolvenKit.

- Add the `.re` animation file you wish to import to your **/Raw** folder in your mod project
- Add the `.anims` animation set file from the game with the *Asset Browser*
- Click View > Import/Export tool
- In the Import/Export tool, navigate to *Import*: a list of all `.re` animation file in your **/Raw** folder will appear
- Double click the entry to change the import options:
  - Select the `.anims` animation set file to import from the dropdown
  - Select the *animation name* you wish to override from the dropdown
- Click *process*

> INFO - It is best to name `.re` animation correctly even if you replace an existing animation, because the imported animation will always have the filename of the `.re` file.

Animation Modding - Manually

> See also: **Import Command**

The `import` command imports an `.re` animation file into an existing `.anims` animation set file.

Optional parameters include specifying an existing animation name to rename inside the animset file (otherwise the filename of the `.re` animation file is used) or specifying a different output path (default is overwrite).

> INFO - It is best to name `.re` animation correctly even if you replace an existing animation, because the imported animation will always have the filename of the `.re` file.

Script Modding

Place modified `.script` files inside `<Cyberpunk 2077>/mods/<name>/scripts`.

The user will have to compile all installed mods into a modded script blob with REDmod before playing the game.

TweakDB Modding

Place modified `.tweak` files inside `<Cyberpunk 2077>/mods/<name>/tweaks`.

The user will have to compile all installed mods into a modded tweak blob with REDmod before playing the game.

Audio Modding

Audio Modding - WolvenKit Integration

The REDmod animation import tool can be used as a Plugin in WolvenKit.

- Add the `.wav` sound files you wish to import to your **/customSounds** folder in your mod project
- Click View > *Sound Modding tool*
- In the Sound Modding tool, choose which to mod from the list of game sound events
- Adjust the parameters
- Click *Save*
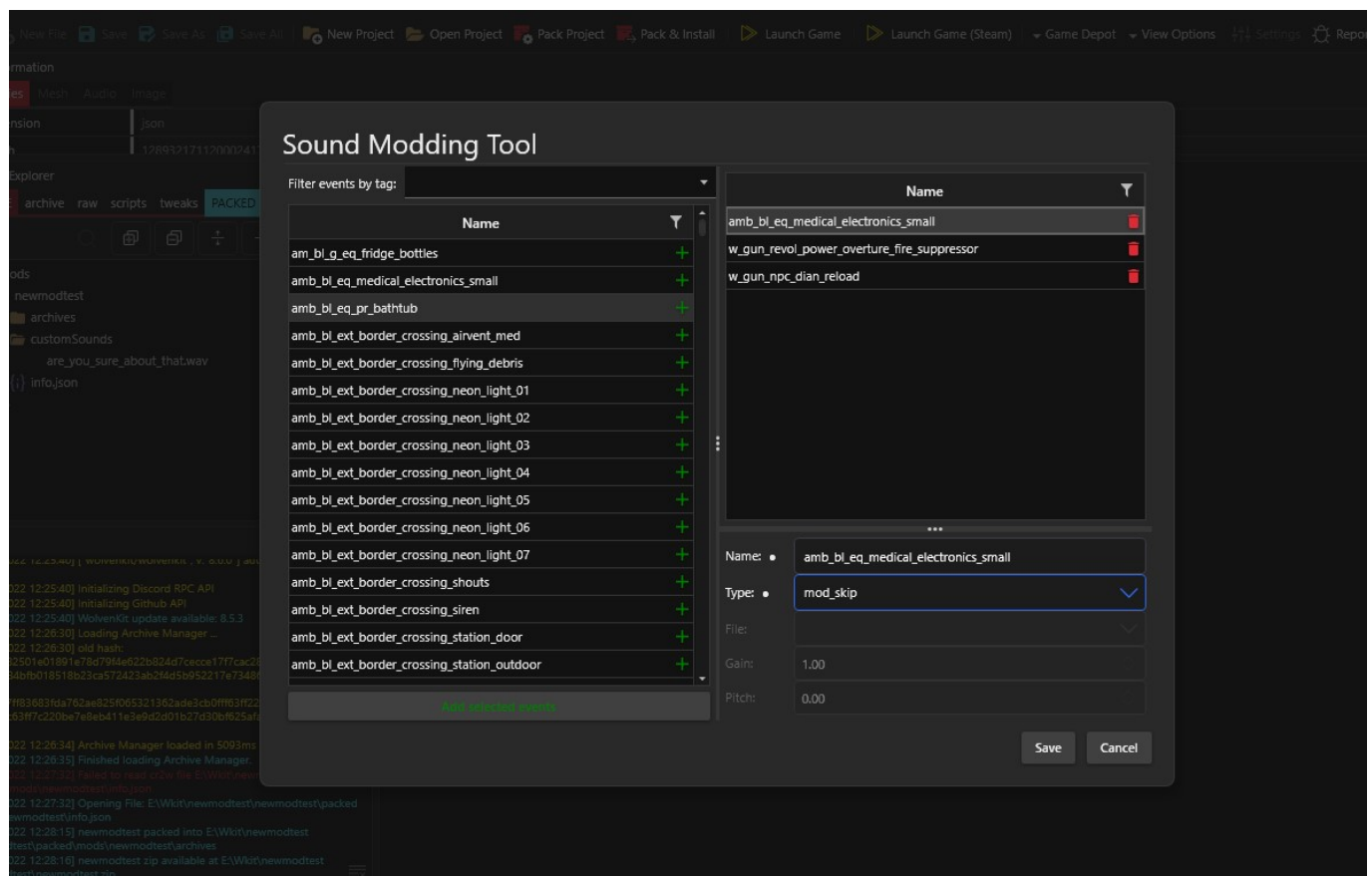
Audio Modding - Manually

Place raw `.wav` audio files inside `<Cyberpunk 2077>/mods/<name>/customSounds`. Include a **info.json** file with your mod (`<Cyberpunk 2077>/mods/<name>/customSounds/info.json`) where you sepcify how to use your custom sounds.

Example

```
{
  "name": "newmodtest",
  "version": "1.0",
  "customSounds": [
    {
      "name": "amb_bl_eq_medical_electronics_small",
      "type": "mod_skip"
    },
    {
      "name": "w_gun_revol_power_overture_fire_suppressor",
      "type": "mod_skip"
    },
    {
      "name": "w_gun_npc_dian_reload",
      "type": "mod_sfx_2d",
      "file": "are_you_sure_about_that.wav",
      "gain": 1.0,
      "pitch": 0.1
    }
  ]
}
```

Parameters

- **name** - the game audio event to override
- **type** - the sound type. Options are:
  - mod_skip: do not play this sound event
  - mod_sfx_2d
  - mod_sfx_city
  - mod_sfx_low_occlusion
  - mod_sfx_low_radio
  - mod_sfx_low_room
  - mod_sfx_low_street
- **file** - the `.wav` file to use (inside `/customSounds`)
- **gain** and **pitch**

Deploy

`deploy` - Deploys mods to the game by compiling them together

Synopsis

```
Usage:
  redmod deploy -root=<path> [options]

Arguments:
  -root=<path>

Options:
  -mod=[<name>]
  -h, --help
```

Description

The `deploy` command stages installed mods to work when the game starts with the `modded` flag.

- archives from `<Cyberpunk 2077>/mods/<name>` are staged to load
- scripts in `<Cyberpunk 2077>/mods/<name>/scripts` are compiled into a modded script blob
- tweak files in `<Cyberpunk 2077>/mods/<name>/tweaks` are compiled into a modded tweakDB blob
- sound files in `<Cyberpunk 2077>/mods/<name>/customSounds` are staged to load

You can specify a specific load order with the **-mod** parameter

Arguments

- `-root=<path>`

  the game root folder for which the deploy command is run.

  | INFO - this needs to be `<Cyberpunk 2077>`

Options

- `-mod=[<name>]`

  only stages specified mods in a comma-separated list

- `-?|-h|--help`

  Prints out a description of how to use the command.

Examples

- **redmod deploy -root="C:\Gog\Cyberpunk 2077"**

  Stages all installed mods under `C:\Gog\Cyberpunk 2077\mods` for loading

- **redmod deploy -root="C:\Gog\Cyberpunk 2077" -mod=modA,modB,modC**

  Stages only the three mods `C:\Gog\Cyberpunk 2077\mods\modA`, `C:\Gog\Cyberpunk 2077\mods\modB` and `C:\Gog\Cyberpunk 2077\mods\modC` for loading

Import

`import` - Import/reimport animset animation

Synopsis

```
Usage:
  redmod import -gameRoot=<depot> -inputPath=<file> -animset=<resource-path>
[options]

Arguments:
  -gameRoot=<depot>
  -inputPath=<file>
  -animset=<resource-path>

Options:
  -outputPath=<resource-path>
  -animationRename=<name>
  -h, --help
```

Description

The `import` command imports an `.re` animation file into an existing `.anims` animation set file.

Optional parameters include specifying an existing animation name to rename inside the animset file (otherwise the filename of the `.re` animation file is used) or specifying a different output path (default is overwrite).

> INFO - It is best to name `.re` animation correctly even if you replace an existing animation, because the imported animation will always have the filename of the `.re` file.

Arguments

- **`-gameRoot=<path>`**

  directory-path to the depot for animset and output (e.g. `C:\modding\depot`)

> INFO - this is the base depot from which relative <resource-paths> are loaded

- **`-inputPath=<file>`**

  absolute path to the animation `.re` file

- **`-animset=<resource-path>`**

  resource .animset file to use

> INFO - resource path must start with `base\` (e.g.
> `base\animations\npc\generic_characters\male_average\open_world\generic_male_averag`
> `e__stand__rh_cane_lh_cigar__01.anims`)

Options

- **-outputPath=<resource-path>**

  resource `.anims` file name to write, default is to overwrite the input animset (resource path must start with `base\`)

- **-animationRename=<name>**

  parameter representing the old name of animation to be replaced by the new animation (default is the filename of the `.re` animation file)

> INFO - the imported animation will always have the filename of the `.re` file even if you specify the animationRename parameter.

- **-?|-h|--help**

  Prints out a description of how to use the command.

Examples

- **redmod import -gameRoot="E:\modding\depot" -input="E:\raw files\animtest1\macarena_long (1).re" -animset=base\animations\npc\generic_characters\male_average\open_world\generic_male _average__stand__rh_cane_lh_cigar__01.anims -output=base\out\new_test.anims - animationRename=stand__rh_cane_lh_cigar__01__smoke__02"**

  Imports the animation file `E:\raw files\animtest1\macarena_long (1).re` into the animset file `E:\modding\depot\base\animations\npc\generic_characters\male_average\open_world\gen eric_male_average__stand__rh_cane_lh_cigar__01.anims` and replaces the animation with the name **stand__rh_cane_lh_cigar__01__smoke__02** with the name **macarena_long (1)**. The result is saved to to `E:\modding\depot\base\out\new_test.anims`.