

Sections

IMPOSSIBLE LEVELS LIST

SHOWCASE AND PLAYTESTING GUIDE

Owned by **Collin and Decks**

Join us on Discord to submit levels: <https://discord.gg/MHx4Nbp>

SHOWCASES

Basically any decent showcase can be added to the list, but if yours isn't really good quality, it can be easily replaced by a better one. If you want to make your showcases good and make sure they stay on the list, there are some guidelines you should follow. This will show you how to make a near irreplaceable showcase for both version 2.1 and 2.2.

2.1 Showcases:

Before Starting:

If you want to showcase a level in version 2.1, be sure to have a downgraded instance of GD installed.

To do this:

1. Open steam://open/console (in a browser or using win+r)
2. Paste into the console: download_depot 322170 322171 641118531549408775
3. Go to /steamapps/content/app_322170/depot_322171 (add after wherever your steam install is)
4. When its finished, create steam_appid.txt and write 322170 inside of it (opens the game via the .exe without adding it to steam)

NOTE: If you do not already have it installed, be sure to install ffmpeg and set it up correctly. Most bots should have it built in, however it is recommended to also install it yourself once. You can use this [video](#) as a guide on how to do so.

Renderer Settings:

Make sure all your showcases are in 4K 60FPS at a bare minimum.

If you are not using a bot with a good built-in internal renderer (ex: Mega Hack Replay), watch this [tutorial](#) on how to use ReplayBot's internal renderer. TASbot and Echo have built in renderers for you to use.

If you do not have ReplayBot, you can install it [here](#).

These are the best settings for 4K:

Resolution: 3840 x 2160

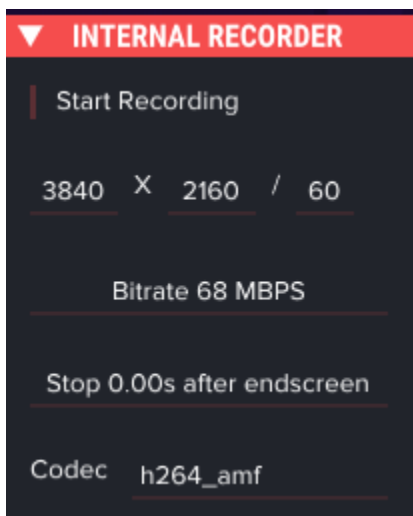
FPS: Any multiple of 60

Bitrate: At least 68mbps

Codec: This document highlights which codec is best for which gpu as well as recommended extra args [Comprehensive FFmpeg Guide for Cosmo](#)



Example of the correct settings using replay bot renderer



Example of the correct settings using TASbot internal renderer

Scroll Speed Bug (SSB) Fix:

If your showcase is of a level with a high fps or is very long, you will get song desync due to a bug known as scroll speed bug, to fix the audio desync, you can use the recorder speed fix provided in TASBot (if you have deluxe) and Echo, you can also edit the audio in a video editor to make it sync or you can use auto music sync in Megahack v7 or GDHM and record the audio using a program like Audacity, lag may be an issue however.

Glow Fix:

Make sure you use glow fix, this is a tutorial on how to fix scaled glow in the game.

1. Download and install [hxd](#).
2. Open HXD and drag GeometryDash.exe into the application.
3. Press ctrl+g and then type "3c1f7", replace the 06 with 00.
4. Download and replace the same file in the Geometry Dash resources file with [this](#) one.

2.2 Showcases:

Before Starting:

Install [Geode](#) to install bots that will be used to create macros and render showcases.

To install other versions of 2.2 for specific bots, use the same steps but modify step 2 to be:

- download_depot 322170 322171 7903404280228366327 for 2.206
- download_depot 322170 322171 2530486154587189554 for 2.204

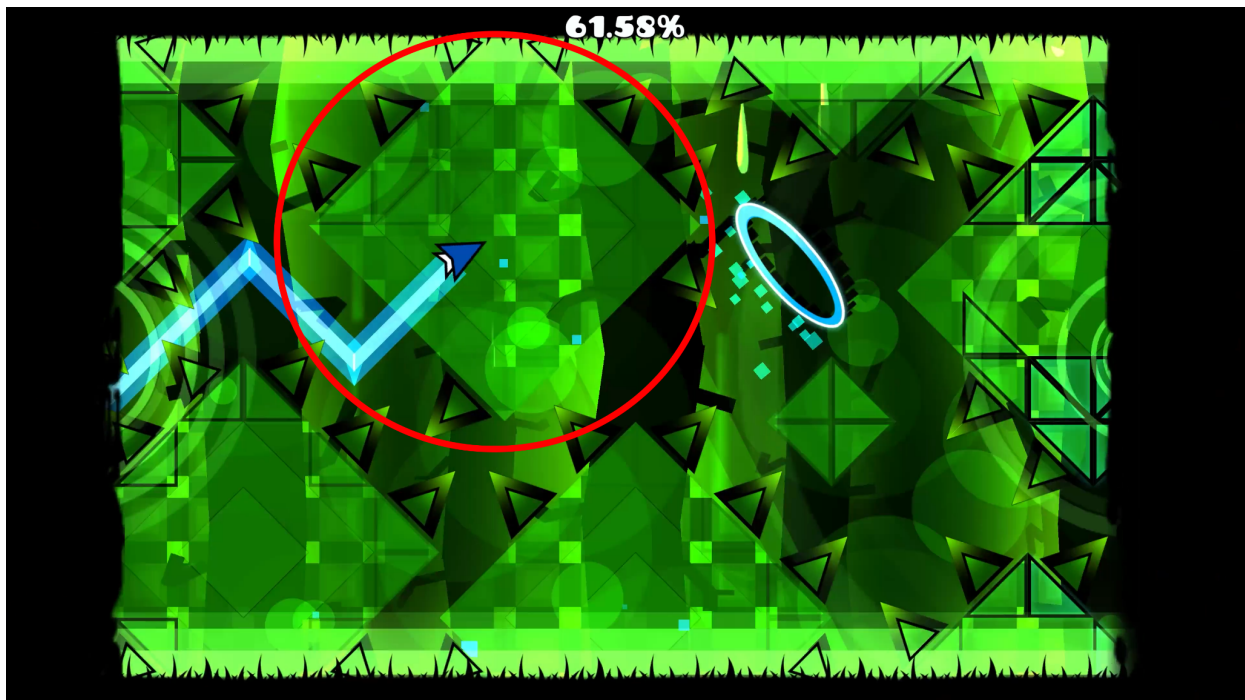
The section regarding **Renderer Settings** still applies the same to 2.2 showcases, **Glow Fix** is no longer an issue as it is fixed in vanilla 2.2. **SSB** is still an issue, however there is no publicly available fix that currently exists for 2.2. This section will be updated once an easily accessible 2.2 SSB fix is released.

General Notes:

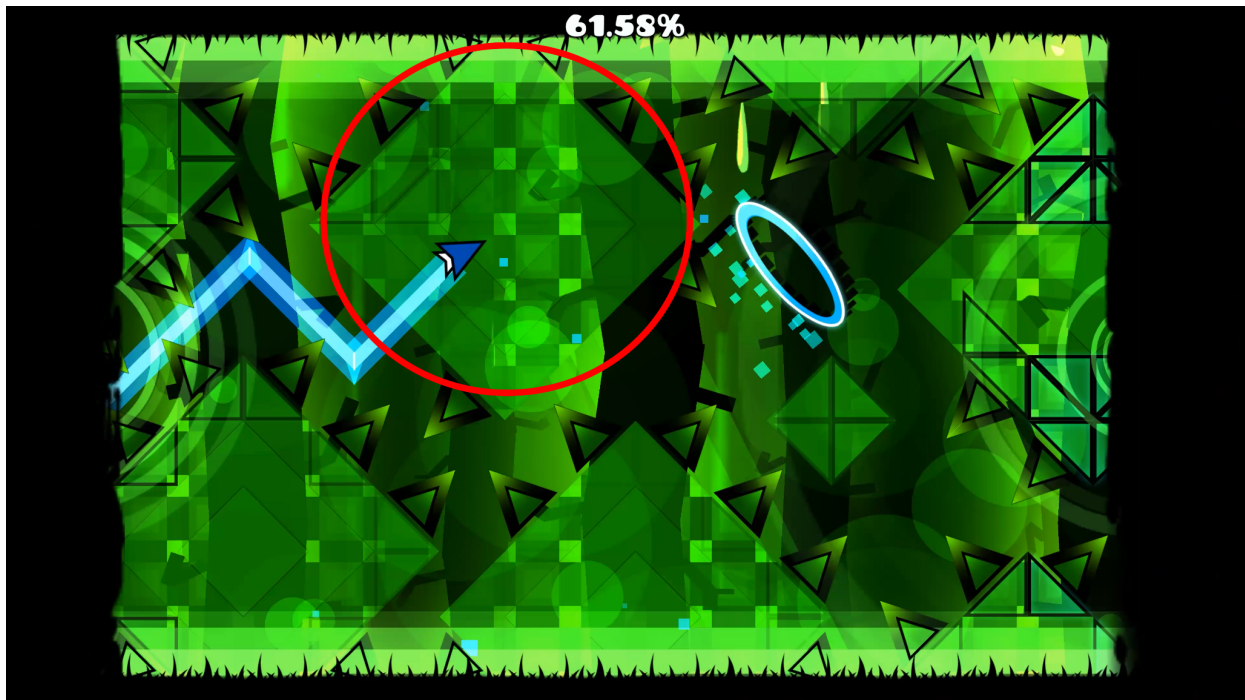
Colour fix:

Typically, renderers don't use the correct color channels when rendering which results in your showcases not having enough green.

Color fixed showcase:



Not color fixed:



Notice how the circled area in the 2nd picture looks darker than the top one.

There are several methods on how to fix the colors in your showcases. This guide will go over the 2 best methods.

- **The Renderer Arguments Method**

Use these if the renderer you are using supports vf arguments.

`colorspace=all=bt709:iall=bt470bg:fast=1`

- **NVI Method**

If you are using a renderer that does not support vf arguments, render your showcases normally first but then go to the folder where your video is saved and you press shift and right click at the same time.

Open Windows Powershell and paste this command with the .mp4 parts being renamed to the name of the showcase file to what the color fixed showcase should be called.

`ffmpeg -i example.mp4 -vf colorspace=all=bt709:iall=bt470bg:fast=1 -b:v 68M -b:a 320K output.mp4`

For example: `ffmpeg -i Slaughterhouse.mp4 -vf colorspace=all=bt709:iall=bt470bg:fast=1 -b:v 68M -b:a 320K Slaughterhousefixed.mp4`

General Rules

- Don't use a texture pack that changes object textures.
- Make sure you have trail bug fix enabled.
- Don't have any labels (except percentage or level time) enabled.
- If the level makes gameplay hard to see, include a layout+hitboxes showcase, within the main showcase or in the description.
- Have most if not all particles enabled as well as wave audio sync and orb audio sync.

Macro Quality:

Make sure your macros are good quality and look human. The more human a macro looks, the better. In general you want to avoid overly precise straight-fly, unnecessarily high cps spamming, unnecessary inputs and anything else that might look inhuman to a casual viewer.

Example of a [bad macro](#)

Example of a [good macro](#)

Extra Tip:

Change the background and ground colour to black when making a layout mode showcase to reduce file size. (This does not affect showcase replaceability)

What makes a showcase replaceable

- The showcase has one of these major issues:
 - A bad macro
 - The showcase is not 4K
 - The showcase is not native 4K (upscaled from 720p or 1080p)
 - The showcase has very noticeable fluctuations in video or audio quality
 - The showcase uses noclip
 - The showcase does not use a creator-intended route
 - The showcase has noticeable scroll speed bug issues
 - The showcase is of an outdated level version
 - The showcase uses a texture pack that changes in-level assets in any way (except glow fix)
 - Bad audio quality
 - No shaders (2.2 only)
 - No camera move or zoom (2.2 only)
 - No in-game sfx (2.2 only)
 - No layout + hitboxes showcase for heavily obscured or memory-reliant levels
 - No [wave trail fix](#) (2.2 only)
- The showcase has multiple (3 or more) small issues, including:
 - No particles
 - No wave pulse
 - 'Trail always on' enabled
 - No pulses / Excessive pulses
 - Unnecessary labels enabled
 - No colour fix
 - No portal effects
 - No percentage label
 - Aspect ratio that isn't 16:9 (unless required by the level)
 - No glow fix (2.1 only)

As long as the showcase does not violate any of the major issues listed (the ones before the small issues), showcases may also be replaced if a specific showcase is requested by the level creator

Following all these steps should result in irreplaceable showcases.

Examples of irreplaceable showcases:

[WNDY](#)

[Locked101](#)

[Xane](#)

[PhiPan](#)

[Zodiac](#)

[Pieguy](#)

[NVI](#)

PLAYTESTING

Playtesting is an important process when making a level for the ILL, it allows you to check that your level is physically possible and meets other gameplay standards like the CPS rules.

Bots

Bots allow you to record a macro of a level in practice mode using tools such as speedhack and frame stepper. This macro can then be played back at full speed as though the level was actually being played by a human. With most bots, you can stop replaying and start recording the macro at any time, so if you find a part that doesn't work, fixing and getting back to it is a simple process. The best bots for playtesting will vary with time and game version; bots that change physics (bots that rely on frame fixes or input fixes), however, will typically deliver mixed results in playtesting, so they are generally better to avoid.

Recommended bots for 2.1 include (in no particular order):

- [Mega Hack Replay](#) (Paid, included in Mega Hack v7)
- [OmegaBot](#) (Free)
- [TASbot](#) (Free and Paid versions)
(Recommend using version 4.20, paired with Mega Hack v7 due to a better practise fix)
- [Echo](#) (Free)

Recommended bots for 2.2 include (in no particular order):

- [Eclipse](#) (Free, included in the Eclipse mod menu)
- [xdBot](#) (Free)
- [ōbot](#) (Paid)
- [Mega Hack Replay](#) (Paid, included in Mega Hack v8)
- [yBot 2](#) (Paid, 2.204 only)

CPS Calculation

Manually testing if a level breaks CPS rules (highlighted in the [rules document](#)) can be more challenging than fixing impossible parts. It involves manually counting all the clicks within a second and calculating the time interval between them depending on the frames elapsed on the set framerate. So, we suggest using a CPS checker that does this process automatically via a macro made using a bot (such as the one included in [this website](#)).

Macro-buffing

Macro-buffing is a term describing buffing the level by placing objects around the path of a macro to buff the level, drastically increasing the difficulty. An example of this is [Domba](#), which had its macro generated beforehand and precisely added objects around it to add difficulty.

If you don't have GDHM Deluxe or are on version 2.2, the best way to macro-buff is to take an input or release and do it with the hitbox trail in Mega Hack or similar menus, then move spikes in as close as possible until it becomes really hard to buff it more, but is still physically possible.

If you have GDHM Deluxe however, you can use its macro buffer to visualize the hitbox trail and you can see if your macro will break, you can see a video of it [here](#).

Macro-buffing is controversial due to its nature and it is seen as unfair forced difficulty by some people, especially with some extreme cases like Domba, where buffs are added all around the trail, leaving little to no room for any deviation.

GLOSSARY

A glossary of common terms you might find in the ILL server:

Est: An est (short for estimate) is a where a person thinks a level in pending should be placed on the list.

Quality Control: Quality Control (QC for short) is a role given to certain members of the server that allow them to vote on levels that are considered for the list. An application has to be submitted to get this role.

Physically impossible - Level is impossible and can't be beaten without noclip even when playing perfectly.

Humanly impossible - Physically possible but impossible for a human to complete.

Bot - Records macros of levels that you can use to record showcases

Macro - A recording of a level that you can make using a bot that almost perfectly plays back what you have recorded.

Physics changing bot - A bot that manipulates game physics and edits a macro so the macro doesn't fail. For example, aBot, oBot and Echo with accuracy fix ++, using those bots leads to inaccurate playtesting.

Xpos limit - the furthest a player can go into a level before it freezes into place, it varies depending on FPS and speed. It usually doesn't matter unless the fps you're using is insanely high or the level is very long. An xpos limit chart for all fps ranges and speeds can be found [here](#).

Scroll speed bug - As you get closer and closer to the xpos limit, your icon can change the speed it's moving at. This causes song desync and may make some levels impossible. An example of the scroll speed bug can be found [here](#).

Color error - This is a problem present in nearly every replaybot and tasbot renderer showcase, it is caused due to imperfections in the recording as the render converts the colormatrix from srgb to bt601 and it makes the showcase look unlike the actual level. The best color fixing methods are found in the showcasing section of this google doc.

Frame gate - A frame gate is a timing that is possible on only one fps value due to the nature of it. For example, the first jump in [OYMYAKON](#) is a frame gate that forces 240 FPS. A video showing how to make a frame gate can be seen [here](#).

Hitbox perfect - It is as the name suggests, a timing or a gap that is very close to, if not perfectly aligned to the player hitbox. These usually result in the timing becoming a frame gate due to the precise alignment needed to pass through.

Pyramid jump - A commonly used type of jump used typically in post 1.7 style impossible levels. It involves the player being a half speed, normal size cube and a pyramid structure consisting of 2 blocks with spikes surrounding them. An example of this can be seen [here](#).

Frame alignment - Imagine a light that flashes 6 times every second and then imagine one that flashes at twice the speed, every 2nd time the faster one flashes, the slower one flashes too. Now imagine a 3rd light that flashes 14 times every second. You'll notice that the third light rarely syncs with the other 2. That's similar to how frame alignment in GD works.

Frame misalignment is when you have a different frame alignment than what you normally would have. You can misalign from several things including entering a portal a certain way or placing checkpoints in midair. A video explaining frame alignment more in depth can be found [here](#).

Swift click - A swift click is a click where you release the same frame you click on, several levels abuse this for difficulty including Formosa. It can also be used to create double clicks, for example the first cube in Formosa. In order to pass it, the player must click and release and then click again, all in the same frame, to jump. Otherwise the player is stuck to the ground due to the multi-activate dash orb.

MINUSwave - A MINUSwave is a form of 2-player wave gameplay where both players go through two separate, very tight wave corridors that have the same structure. Each player has to pass through multiple gravity changes within the corridor, however, the changes are

de-synced to force the two player inputs to deviate from each other. These were popularised by the level [MINUSclub](#), from which the term MINUSwave is derived from.

Contact @wndy_lndscpe on discord for any questions or feedback relating to this document