

# *How to make a Battlezone map without using maketrn*

*By système*

This tutorial is used in the case you don't want or cannot use maketrn.exe. It can come in handy for the case you want to make a duplicate of an already existing map and convert it to another format (singleplayer to multiplayer and vice-versa etc). But it can also be used to make a fully working map from scratch.

## **Step 1: Choosing an ideal bmp template**

Because having to make a map in the Battlezone editor might give too crude results, it is best to start with a bmp generated by an alternate program. For this example, we will use Xaos, a fractal generator (Figure 1). This program is interesting because it is opensource, available on github, and possesses an extended amount of fractals with the option of generating your own, unlike most fractal generators who, too many a time, are focused on the basic Mandelbrot formula.

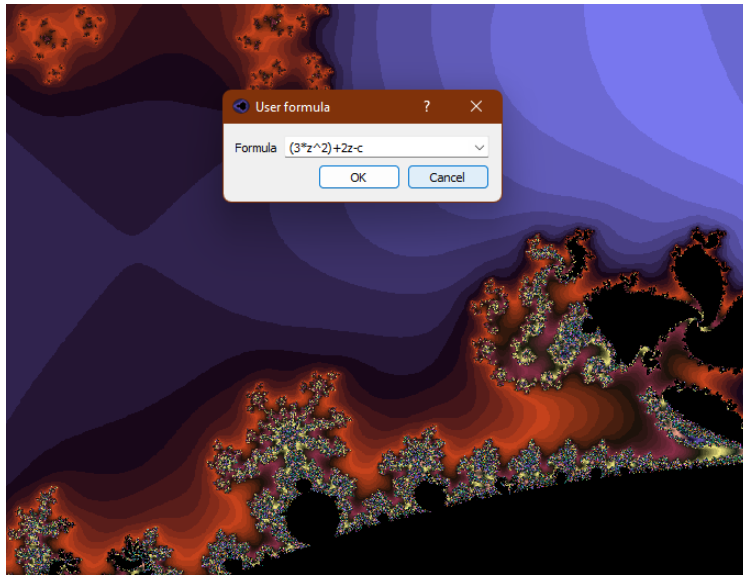


Figure 1: Xaos screenshot showing the user formula menu, along with the obtained fractal representation.

Once your fractal is generated, you could browse a bit through it and zoom on interesting points. Keep in mind that fractals can be zoomed in indefinitely, hence generating as many landscapes as possible (Figure 2). Once you've reached the desired configuration, either save the bmp or take a screenshot. Keep in mind that your bmp's dimension must be a multiple of 128 (128x256 for example)

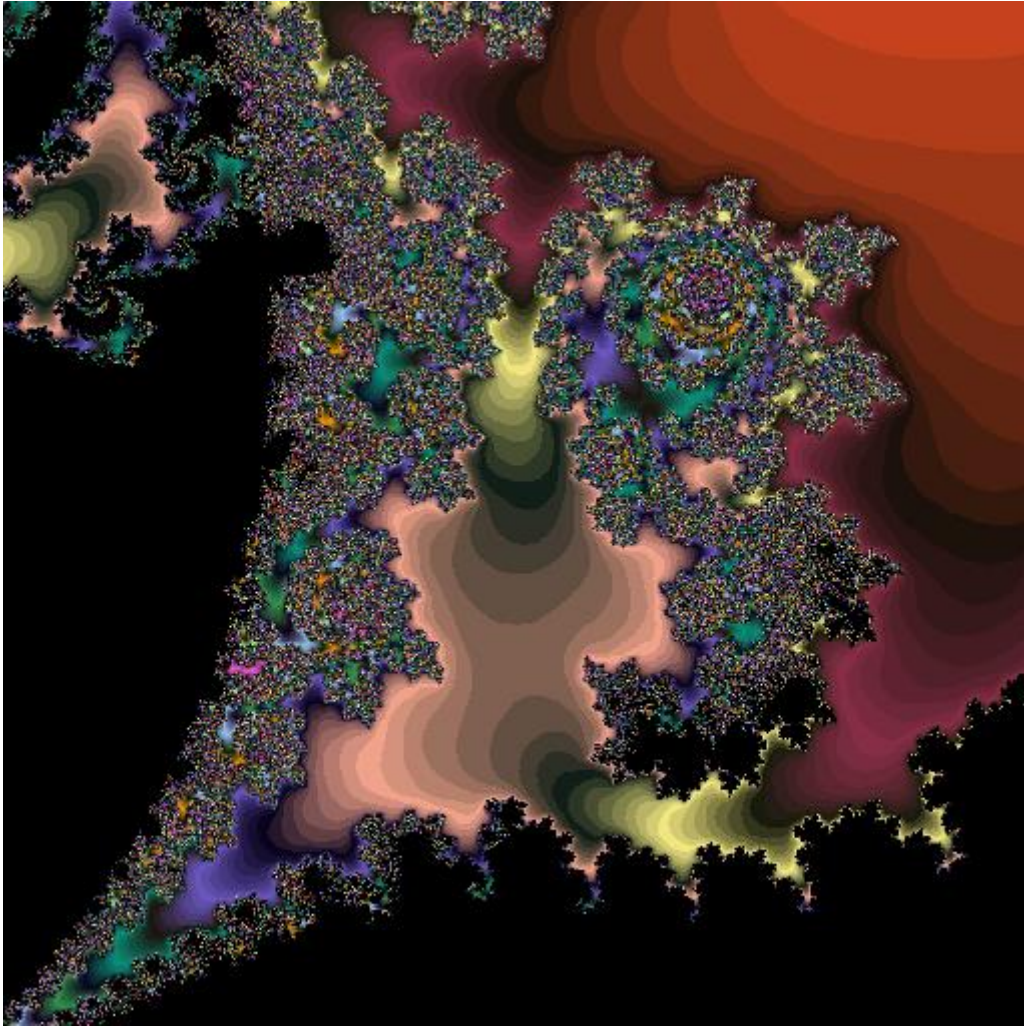


Fig 2: A place of interest within the fractal

## Step 2: Converting your bmp into a Battlezone friendly format

The two common options to convert your obtained bmp into a battlezone heightmap (.hgt format) are `bzhgt` and `bzfrac`. Because `bzfrac` only has a few loading options (averaged, divided, and masked) to load the bmp, it is likely that we lose important height details when porting the map directly. Thus, it is highly recommended to use `bzhgt` first. This latter program possesses a precision coloring mode, a feature of value that will convert tones into heights with 100% precision. This option, however, is listed aside from the general modes and can only be activated by pressing 0 instead of keys from 1 to 5 as it would suggest (Figure 3). To use `bzhgt` correctly, simply drag and drop your map onto the exe.

```
HGT Convertor for BattleZone map makers

c:\users\adell\pictures\bfg.bmp found and loaded

Select Height Modifier: 1(flat) - 5(high)
Or 0 for precision colour mode.

0 selected
4 chunks(s) detected...
Creating chunk ####
File c:\users\adell\pictures\bfg.hgt created sucessfully!

Any key to exit
```

Figure 3: Using precision colour mode.

Whether you used bzhgt or wanted to proceed to load your map directly into bzfrac, it is strongly advised to use the new version, available on the maproom, which does not the Wing32.DLL error (Figure 4). This bzfrac also possesses an US and an EU versions whose sole difference, albeit capital, resides in the fact one uses “.” and the other “,” for decimal numbers.

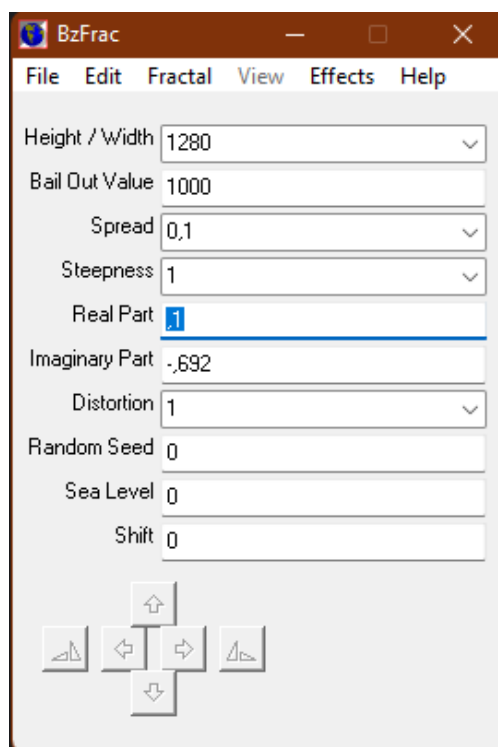


Figure 4: bzfrac\_EU, which uses “,”. It is shown for example in the 0,1 spread.

### Step 3: Some tips about bzfrac

As evidenced by the program, you can open your hgt file by using open directly. Bzfrac is pretty tolerant in terms of supported programs and can also be used to open binary files, bmps and even text tiles. Note that if you want to import a bmp file into bzfrac you have 3 options: averaged, which will try to balance between levels, divided, which does not, and masked, which tries to blend the heights together, avoiding sharp cliffs. In our case, we opened the converted hgt (Figure 5).

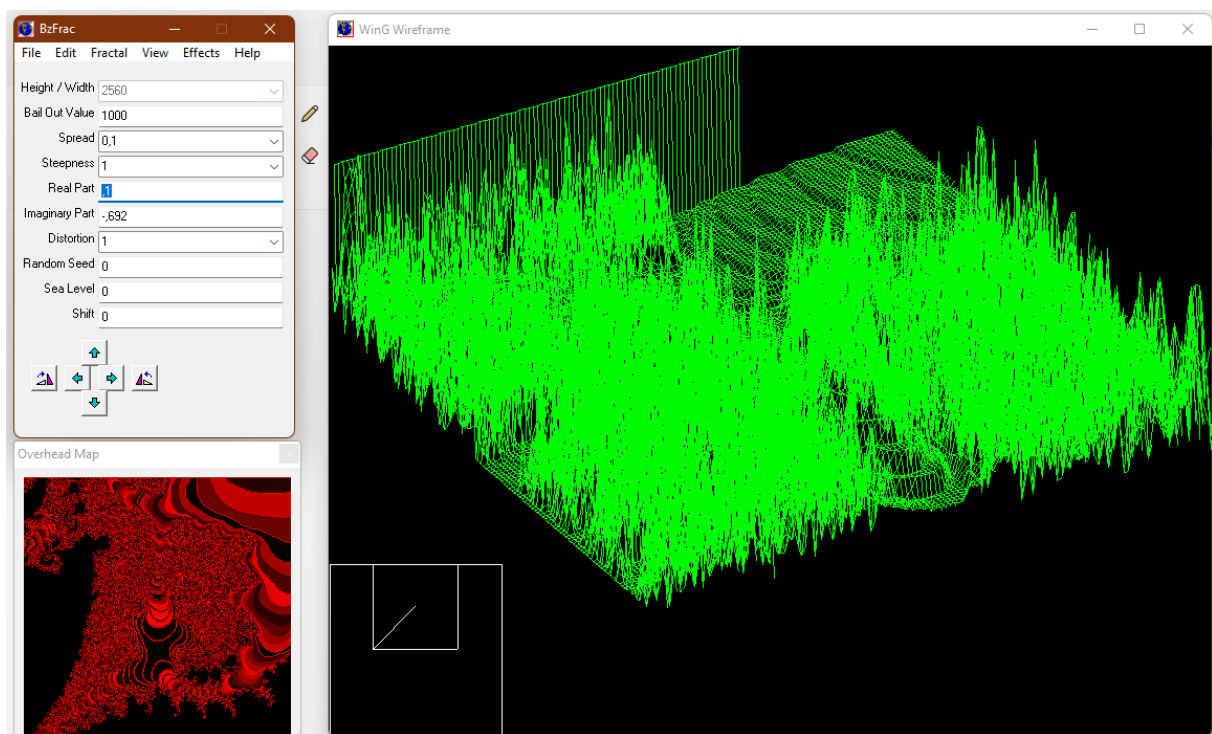


Figure 5: editing a hgt file in bzfrac.

As seen in the figure above, the current heightmap might be too “sharp”. To make the map more AI friendly, we would have to smooth the walls a bit, so they become practicable slopes. For this, there exists a tool in the effects menu that is called smooth. This operates automatically and possesses its own shortcut: ctrl s. the Effects menu has a plethora of tools that can help make the final map more enjoyable without going through the struggle of editing it bit by bit in the editor. Such tools include increasing or decreasing the height, mirroring, adding ripples, volcanos and simulating an earthquake which would cause some erosion on your terrain. For these tools to become effective, you can change the steepness in the main menu (it is currently set to 1 but you can change it to any positive integer).

Additional tip: to give equal chances for all players in the case of a player versus player map, you can try the mirroring option, so your terrain becomes symmetrical.

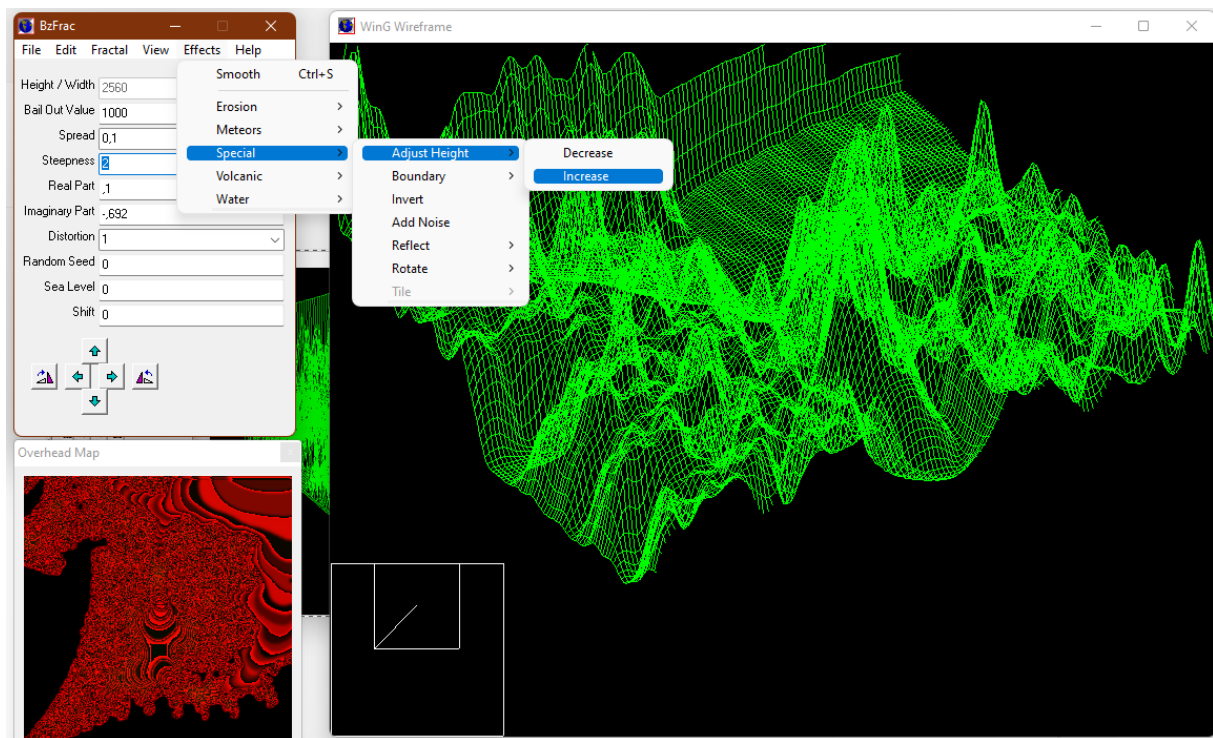


Figure 6: Operating an effect on bzfrac. In this case, we adjust the terrain height

#### Step 4: Giving more tweaks to your map

Because it's sometimes hard to obtain what we want, we're likely to add the most important details manually. To this extent, we can convert the map back to an image format. Unfortunately, the option to save the map as a bmp gives mixed results with bzfrac, so we would be prone to save it as a hgt first, then convert it with bzhgt. Just like we did in step 2, we can drag and drop the hgt on the exe. It will ask us what size we want for the final map, based on the blocks it detects. Once again, it is recommended to choose the actual size of the map, as results might vary. Nonetheless, bzhgt would convert your hgt to a tga file format, which it cannot use. After editing your image, it is best to save it as a 24bit-bmp so it can be read safely by both editing programs presented in this tutorial.

#### Step 5 : Making of an empty mission

Because Battlezone 1.5.2.27u1 only needs the heightmap and the trn to run a map properly, we are already halfway done. Thereby, all we need is a matching trn file. So we can look for a trn which has a size section that matches our hgt (for example, one with a width and depth of 2560 in our case), copy it and name it after our hgt. In this example, we will call both files "thismap" (Figure 7). Keep in mind that battlezone can only handle up to 50 folders inside addon, after what it would not start correctly.

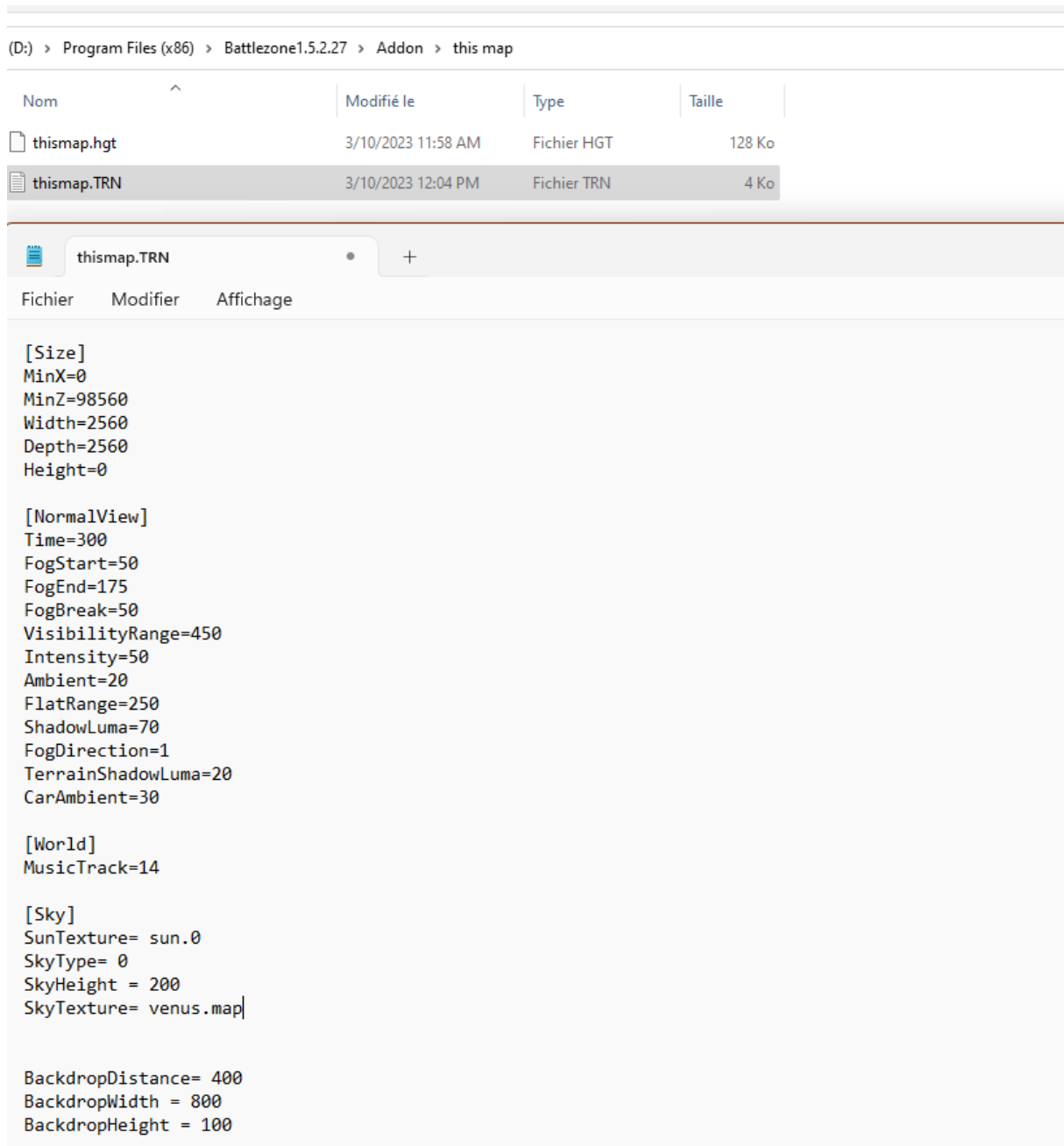


Figure 7: the first step for map generating: creating the hgt and the trn. For clarity, both files are placed in a folder named after the map

## Step 6: Terrain file (trn) tips

While we're at it, let's have a look at the trn and see how we could make our map a bit more original. First of all, we have the Sky parameters. As the name suggests, this section contains all parameters needed for the sky to display properly. First, we have the sun texture. Because the game only has one sprite of the celestial body, it is best to leave it as Sun.0. If your mission is played at night or on a faraway planet, you can leave this entry blank, effectively getting rid of the sprite. Skytexture on the other hand is an actual texture loaded on the higher section of

the sky. Right under it is the Stars section, which contains almost every sprite element displayed in the sky. Generally, each of them has parameters matching the order they appear in the trn. In this example, we only have 1 sprite, but since it's the first, its properties will be followed by 00, if we had another one it would be 01 etc. Texture is the image that would be displayed. Size, as its name suggest, is the size of the texture. Azimuth is the angle formed by its position and the first point on an imaginary circle it is on (for instance, we place it at 180, which is 180°, making it appear opposite from the origin point). Elevation is the height of the element in the sky (15m here) and roll is how it is rotated. By putting 0, it would be displayed exactly as shown in its file, if we put 90, it would be rotated by 90° and so on. Starlist is in case you need actual stars in your map. stars.map is the standard texture and the only one available in the game.

```
[Stars]
Radius      = 300

Texture00   = Neb1.map
Size00      = 200
Azimuth00   = 180
Elevathln00 = 15
Roll00      = 0

[StarList]

Solid       = 0
LayerHeight = 15
Texture     = stars.map
Radius      = 1000

Start0      = 235
End0        = 510
Size0       = 350
Count0      = 12

Start1      = 235
End1        = 510
Size1       = 350
Count1      = 9

Start2      = 0
End2        = 360
Size2       = 350
Count2      = 11

[Color1]
```

Figure 8: the star parameters.



Now, to give the actual sky the desired appearance, we have to choose a color palette. A list of palettes already exist within the game data, with each name matching its planet. Luma, translucency and alpha were used in software mode and would have little impact in 1.5 and above. Changing them is discouraged.

```
[Color]
Palette=VENUS.ACT|
Luma=venus.LUM
Translucency=venus.TBL
Alpha=venus.ALB

[World]
MusicTrack=3
```

Figure 9: Color and music tracks

## Step 7: Loading the map in Battlezone

A crucial step in mapmaking, opening the map in battlezone is key to generating the .bzn, .mat and .lgt. To do this, we have to make a to battlezone.exe and change the target properties as in figure 10.

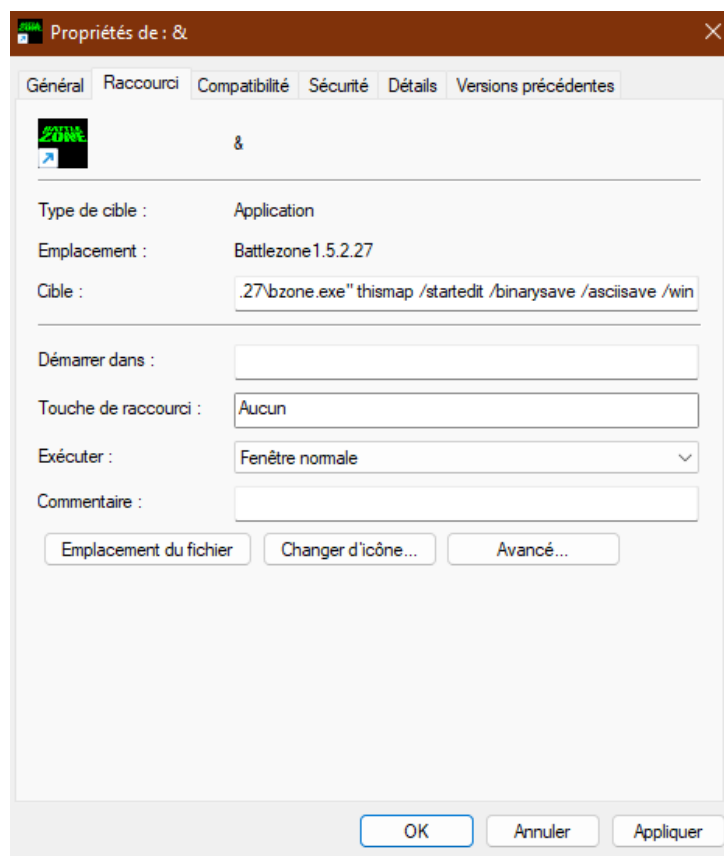


Figure 10: Editing the file properties of the shortcut to load our map.

At this point, all is left to do is to double click the shortcut to see whether the game was able to open your map and generate the missing files. If the .hgt is reasonable and matches the defined dimensions in the .trn, no problems should occur. Please note that, regardless of the files or their nature, an error message will show up because we're not loading the bzn but a virgin map. This is a recurring and normal process and originates from a glitch in 1.5 that is pending a fix. Click ok and proceed to let the game load. Once done, you should be seeing your map properly loaded in the game (Figure 11).



Figure 11: The map generated according to the .hgt and properties set in .trn

### Step 8: Empty mission?

So far, we loaded our map without trouble. We now have to save it (by pressing ctrl s in-game) to obtain the .bzn. the .bzn is a mandatory file that saves the objects that were placed within the map, their status, health and all the various properties they might have. But that is not all. The .bzn also defines what is the type of your map. is it a singleplayer mission, a lua mission, a multiplayer mission and if that's the case is it a strategy map or a deathmatch? Is it none of the above? To clearly define the mission type, we have to look in that very .bzn file. Luckily, as we have defined in the shortcut, it is saved in ASCII, which is readable by humans, the

game normally saving in a file format that isn't really that easy to figure. All what's left to do then is to open the .bzn in notepad and look at the bottom of it. We should find the following word: "EmptyMission". This is because the map was generated by the game and it is set as the default map type, or empty mission. But, not to worry, that can easily be changed to the map type we desire.

```
asuser
perceivedTeam [1] =
1
name = EmptyMission
sObject = 0C1410D0
[AiMission]
[AOIs]
size [1] =
0
[AiPaths]
count [1] =
0
```

Figure 12: Changing EmptyMission to a valid mission format

To have an optimal mission type we could use one of the following, according to what the map needs to be:

- Luamission (is a mission with a Lua)
- Inst4XMission (standard Instant Action format, now supports Lua)
- MultDMMission (for death match maps)
- MultSTMission (for strategy maps)
- EmptyMission (default map, does not handle Lua and is devoid of win or loss condition, which can be handy for editing since other map types would end after a short delay due to the lack of AI units)

Other map types exist, but the first five are the most commonly encountered and are sufficient for most cases. Keep in mind that an MPI is just a MultSTMission with a lua file.

If you have reached this point, it means that you have a working map. Congratulations. Don't hesitate to look at other available mapmaking tutorials and `mapedit.doc` for additional information regarding map editing in `battlezone1.5.2.27u1`.

Sources used:

Xaos, available in <https://github.com/xaos-project/XaoS/releases>

BZHGT, available in <https://bzmaps.net/misc.php#bzhgt>

BzFrac\_EU, available in <https://bzmaps.net/misc.php#bzfrac>

Thismap, a map generated specifically for this tutorial, available in [https://bzmaps.net/misc/tutorials/this\\_map.zip](https://bzmaps.net/misc/tutorials/this_map.zip)