

# Praetorians map editor manual

Imperial Studios 2008

---

<b>I. Introduction .....</b>	<b>3</b>
<b>II. Getting started .....</b>	<b>4</b>
Downloading the map editor .....	4
Installing the map editor .....	4
<b>III. Understanding projects .....</b>	<b>6</b>
XML files .....	6
PVE files .....	7
PTE files .....	7
MOB files .....	7
MLG files .....	8
PRA files .....	8
H2O files .....	8
MSS files .....	8
TGA files .....	8
Creating a new project .....	8
Editing an existing project .....	9
<b>IV. Terrain.....</b>	<b>11</b>
Heightmaps .....	11
Colormaps.....	12
Tiles.....	12
Layers .....	12
<b>V. Creating terrain.....</b>	<b>14</b>
Calculating the dimensions .....	14
Calculating the elevation.....	14
<b>VI. Configuring the editor .....</b>	<b>15</b>
Media folders .....	15
The export path.....	15
<b>VII. The graphical user interface .....</b>	<b>16</b>
Using brushes.....	16
Using layers.....	17
Painting tiles .....	18
Adding tile logic flags.....	18
Adding tile types.....	18
Painting grass .....	18
Drawing water.....	18
<b>VIII. Finishing your project .....</b>	<b>20</b>
Testing your map .....	20
Packaging your map .....	20
<b>IX. Common problems .....</b>	<b>21</b>
My chief cannot enter a village! .....	21
I cannot build a garrison!.....	21
Why is there is no sound?.....	21
Why is there is no rain, thunder, and lighting in the game! .....	21
<b>X. Keyboard shortcuts .....</b>	<b>22</b>
<b>XI. Conclusion .....</b>	<b>23</b>

=====

## Introduction

---

This map editor is part of the MOD Project by Imperial Studios. You can find more information about the project by visiting <http://mod-project.com>. See the forum if you need help on using their software.

In this manual, you will start by learning about the basic requirements and how to properly install the editor. You will learn how to create new projects and edit existing projects. You will learn about an alternative way of creating terrain that involves drawing on a 2D monochrome image. You will learn how to adjust the map editor to suite your own needs by changing its configuration. You will learn how to use its graphical user interface. Finally, you will learn how to test and package your map.

=====

## Getting started

---

Since the editor does not include textures for your terrain, it depends on Praetorians for them. Though having Praetorians installed is not necessary, it is recommended because the game has many other useful resources besides terrain textures.

### Downloading the map editor

The map editor's project website address is <http://www.assembla.com/wiki/show/Cartography3D>. That website was setup by the developers. It is where they kept the technical stuff and discussed things like cracking and program bugs. So, you will not only find the latest version of the map editor from there, but also its code and a few technical documents (project closed - September 09, 2008).

### Installing the map editor

Since the editor comes in a ZIP file, installation is as easy as unzipping the file into a directory. If you have Praetorians installed, then you can unzip everything into the game's folder named *Praetorians* (see image 1).

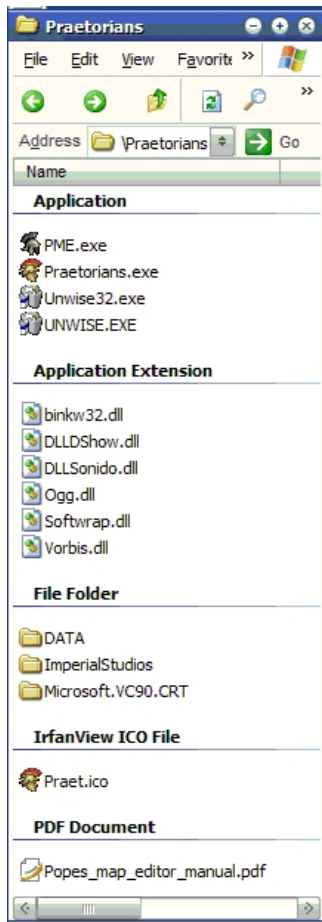


image 1

## Understanding projects

---

Your projects will be split into several different files:

- PVE
- PTE
- MOB
- MLG
- PRA
- H2O
- MSS
- TGA

Splitting a project this way instead of packing everything into one huge file makes it easier to change things. For example, you could have two projects that share files, but differ slightly in some way. These files, together, make up a map and are used by Praetorians. The only file that you *might* ever modify is the MSS file, which can be opened and edited using Microsoft Notepad.

### XML files

Each file in a project is linked together in an [XML document](#) to form a project. XML stands for Extensible Markup Language. It is a generic framework for storing any amount of text or any data whose structure can be represented as a tree. Its primary purpose is to help information systems share structured data. There are two documents, each describing a specific group of maps: projects (*plantillas.xml*) and finished maps (*mapas.xml*).

=====

Here is an example of a structured XML project document:

```
<Plantillas>

  <Plantilla name = "my project template">
    <Minimap name = "myproject.tga" />
    <Texturas name = "myproject.ptc" />
    <Visual name = "myproject_height.bmp" />
    <Colores name = "myproject_color.bmp" />
    <Mode name = "1v1" />
    <Size x = "140" y = "120" />
    <MaxPlayers x = "4" />
  </Plantilla>

  <Plantilla name = "pyros template">
    <Minimap name = "MP_4_1.TGA" />
    <Logica name = "MP_4_1.MLG" />
    <Objetos name = "MP_4_1.MOB" />
    <Texturas name = "MP_4_1.PTE" />
    <Visual name = "MP_4_1.PVE" />
    <Praderas name = "MP_4_1.PRA" />
    <Agua name = "MP_4_1.H2O" />
    <Mode name = "2v2" />
    <Size x = "140" y = "140" />
    <MaxPlayers x = "4" />
  </Plantilla>

</Plantillas>
```

A project document like this tells the editor which file belongs to which project. This is how a project is setup.

## PVE files

These file contains data for a terrain's dimensions, height, shading, and tiles.

## PTE files

These files contain the terrain textures. You can only use copies from the game, but if you know how to use a hex editor, then you can probably create your own.

## MOB files

These files contain data that indicates where objects, such as trees or structures, are positioned on a map.

## MLG files

These files contain data that describes the terrain ground. There are different types of ground, such as sand, snow, water, etc., and each type of ground has certain properties. For instance, water is not passable by legionaries.

## PRA files

These files contain the data that describes grass areas.

## H2O files

These files contain data that describes water areas.

## MSS files

These files describe the map by indicating the position and orientation of villages, critters, the camera, and sounds.

## TGA files

These are image files that are used by Praetorians as minimaps.

## Creating a new project

You have seen what the XML project document looks like. Starting a new project is as simple as adding new data similar to what is already in the XML document. Lets start an example project with a map is similar to "Crossroads". We want its dimensions to be 130 by 130, we want the same textures as "Crossroads", and we want to name it "noob map".

=====



Here is what we would do:

```
<Plantillas>

  <Plantilla name = "noob map">
    <Texturas name = "MP_4_3.PTE" />
    <Mode name = "2v2" />
    <Size x = "130" y = "130" />
    <Elevation max = "2.98" min = "-0.75" />
    <MaxPlayers x = "4" />
  </Plantilla>

  <Plantilla name = "greyscale image template">
    <Minimap name = "myimage.tga" />
    <Texturas name = "myimage.pte" />
    <Visual name = "myimage_height.bmp" />
    <Colores name = "myimage_color.bmp" />
    <Mode name = "1v1" />
    <Size x = "140" y = "120" />
    <MaxPlayers x = "4" />
  </Plantilla>

  <Plantilla name = "pyros template">
    <Minimap name = "MP_4_1.TGA" />
    <Logica name = "MP_4_1.MLG" />
    <Objetos name = "MP_4_1.MOB" />
    <Texturas name = "MP_4_1.PTE" />
    <Visual name = "MP_4_1.PVE" />
    <Praderas name = "MP_4_1.PRA" />
    <Agua name = "MP_4_1.H2O" />
    <Mode name = "2v2" />
    <Size x = "140" y = "140" />
    <MaxPlayers x = "4" />
  </Plantilla>

</Plantillas>
```

But how did I know to use MP\_4\_3.PTE? Because I looked in the *Mapas.xml* file under "Crossroads". Now, when you press the "New map" button in the editor, your project will appear as a new item. If you are wondering about "Elevation", don't worry about it now. You will learn how to use that soon.

## Editing an existing project

Lets say that you had started a new project and exported it, but left it unfinished. Before you can start updating it, you must add some data to the *plantillas.xml* file. Remember, each file in a project is linked together in the XML file to form a project. So, what we are supposed to do is link our exported files to our project.

=====

Here is a modified version of the XML document example from above:

```
<Plantillas>

  <Plantilla name = "noob map">
    <Logica name = "noob map.MLG" />
    <Objetos name = "noob map.MOB" />
    <Texturas name = "noob map.PTE" />
    <Visual name = "noob map.PVE" />
    <Praderas name = "noob map.PRA" />
    <Agua name = "noob map.H2O" />
    <Mode name = "2v2" />
    <Size x = "130" y = "130" />
    <MaxPlayers x = "4" />
  </Plantilla>

  <Plantilla name = "greyscale image template">
    <Minimap name = "myimage.tga" />
    <Texturas name = "myimage.pte" />
    <Visual name = "myimage_height.bmp" />
    <Colores name = "myimage_t_color.bmp" />
    <Mode name = "1v1" />
    <Size x = "140" y = "120" />
    <MaxPlayers x = "4" />
  </Plantilla>

  <Plantilla name = "pyros template">
    <Minimap name = "MP_4_1.TGA" />
    <Logica name = "MP_4_1.MLG" />
    <Objetos name = "MP_4_1.MOB" />
    <Texturas name = "MP_4_1.PTE" />
    <Visual name = "MP_4_1.PVE" />
    <Praderas name = "MP_4_1.PRA" />
    <Agua name = "MP_4_1.H2O" />
    <Mode name = "2v2" />
    <Size x = "140" y = "140" />
    <MaxPlayers x = "4" />
  </Plantilla>

</Plantillas>
```

You only have to do this once for each new project. You can also exclude parts. For instance, if you don't want to import the water file, then just delete `<Agua name = "noob map.H2O" />`.

=====

## Terrain

You don't need to be a computer programmer to understand what virtual terrain really is and how it is simulated by a computer. If you follow along carefully you will see that it is not as technical as it seems.

Terrain is constructed out of points which are located in 3D space. These points are connected to form triangles and these triangles are then covered with a texture. Thus, the terrain might appear as a solid object, but it is in fact a collection of many small textured triangles (*see image 2*).

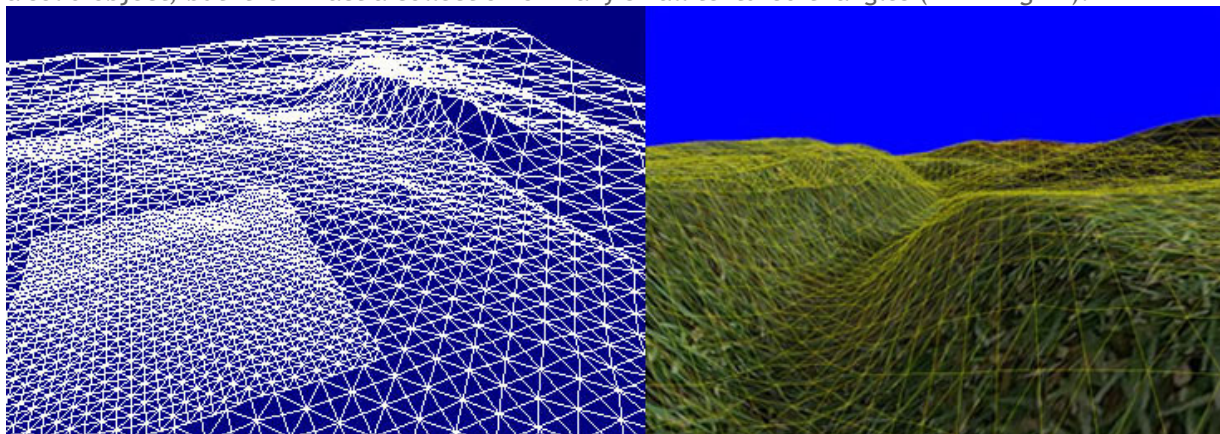


image 2

## Heightmaps

The standard way of creating 3d terrains is through the use of heightmaps. A heightmap is simply a 2d array of values. Each value in the array represents the height of the terrain at that value's position (*see image 3*).

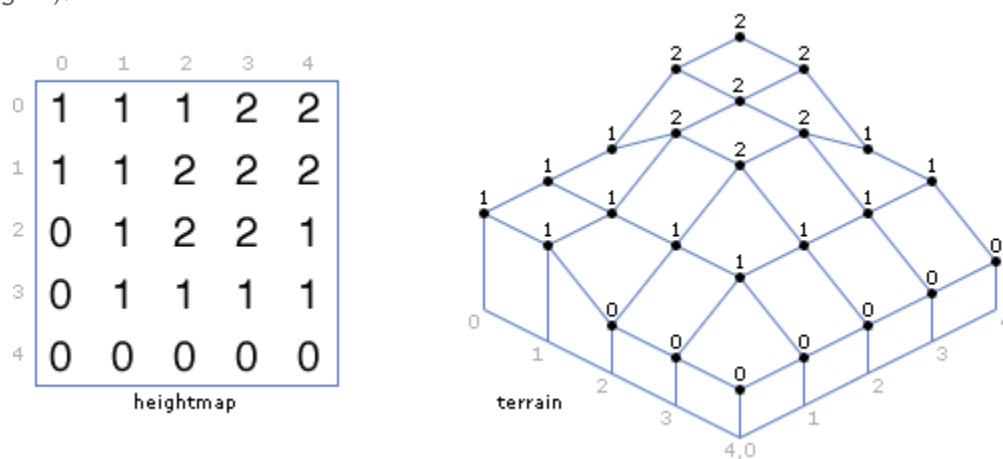


image 3

Another way to imagine a heightmap is to think of it as a [greyscale image](#), where the brightness of each pixel corresponds to the height of the terrain at that point (*see image 4*).

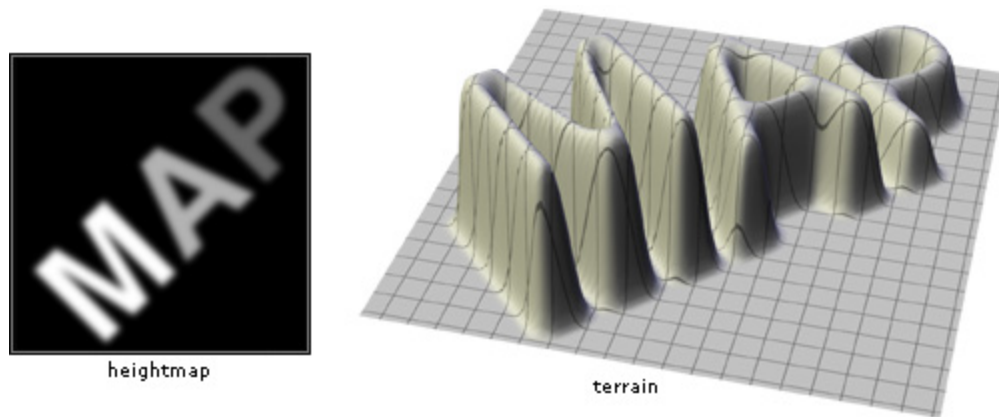


image 4

## Colormaps

A colormap is the same as a heightmap, but its values are used differently. Your terrain will, of course, need dark and light areas. You want to give it the appearance of shadows. This is where the colormap is used.

## Tiles

A tile is a square piece of the terrain (*see image 5*). It is simply a way of organizing data. Do you remember what an MLG file is? It describes the type of ground: sand, snow, water, etc., and its properties. So, the MLG file is essentially describing tiles. A tile can be of only one type, but can have many different properties.

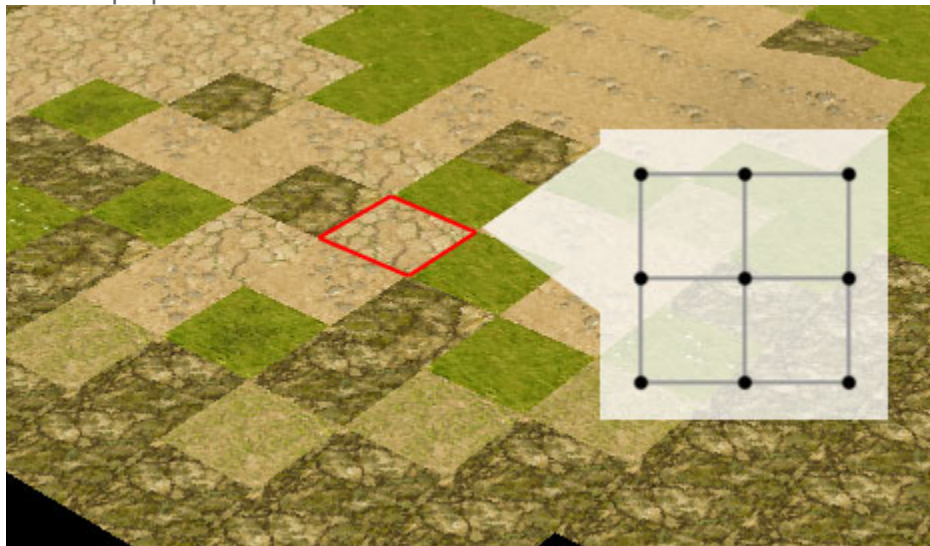


image 5

## Layers

In the editor, you will be working with many different layers, like in Photoshop. There are layers for nature, animals, players, villages, structures, water, grass, and there are two layers for tiles. But

why two layers for tiles? The answer is simple: you will be blending the top layer with the bottom layer to make the terrain look realistic.

=====

## Creating terrain

---

You have two options for creating terrain: using the editor or using greyscale images. Using greyscale images is the easiest and fastest way. You can use any image editing software to make your greyscale images. There are even online programs that will create them for you. The editor will work only with 8-bit greyscale bitmaps. Two important things to remember when editing your images are that they must be edited in "rgb" mode and saved in "greyscale" mode.

### Calculating the dimensions

When creating terrain using greyscale images, you will need to first determine what the dimensions of the terrain will be. You will then use the values to calculate the image dimensions. To do that, we use a formula:

$$\text{width in pixels} = (\text{number of tiles east and west}) * 2 + 1$$

$$\text{height in pixels} = (\text{number of tiles north and south}) * 2 + 1$$

For example, the map "Crossroads" has terrain dimensions that are 130 by 130, or 130 tiles by 130 tiles. That means the heightmap image used to create the terrain would be 261 by 261.

### Calculating the elevation

The elevation determines the steepness of hills on the terrain. Your terrain will have minimum and maximum elevation values. In other words, the minimum and maximum height of the terrain will be above or below zero along the Y-axis. There is no standard way of calculating the minimum and maximum. The values used in the map "Crossroads" are:

max: 2.98

min: -0.79

A greyscale image allows 256 different intensities (i.e., shades of gray). By using this along with our min and max elevation values, we can determine what each pixel of the image translates to in 3D:

$$\text{3D height value} = ((\text{max} - \text{min}) / 255) * \text{pixel\_color} + \text{min}$$

=====

# Configuring the editor

The editor can be configured to suite your own needs. There are many options to choose from like enabling or disabling background music, whether its full-screen or not, the window dimensions, and its appearance. You can even design your own theme for the graphical user interface. All of these can be changed in an XML file. The common settings that users change can be found in the *stage.xml* file. If you don't want the editor in full-screen mode, then change its value to "false". If you want to enable v-sync, then change its value to true. Its that simple.

## Media folders

A media folder is a directory. The editor will search a media folder for the resources its needs (e.g., sounds, images, XML files).

## The export path

This is the same as a media folder, but it is also used as a place where your projects are saved.

## The graphical user interface

The interface might seem intimidating at first glance, but once you start using it you will soon realize that it wasn't worth bitching about. It's so easy, a caveman can learn to use it (see *image 6*).

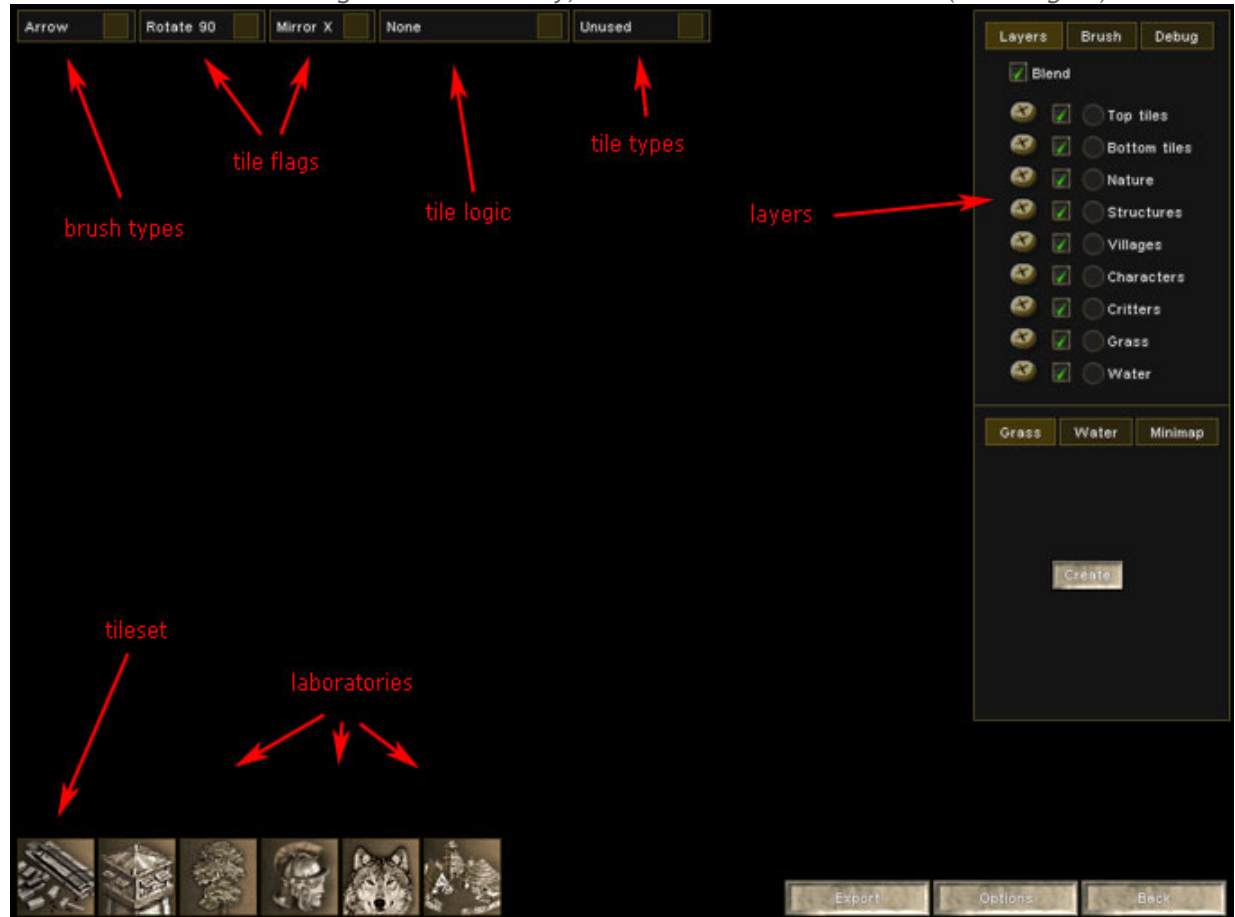


image 6

## Using brushes

A brush has two important attributes: its mode and its type. The mode determines how it is used. There are many different types, but only 4 modes of brushes:

- tile
- grass
- water
- model

When you are working with tiles, you switch to tile mode; when you are working with grass, you switch to grass mode; and so on. You can select the brush mode in the "brush tab" (see *image 7*).



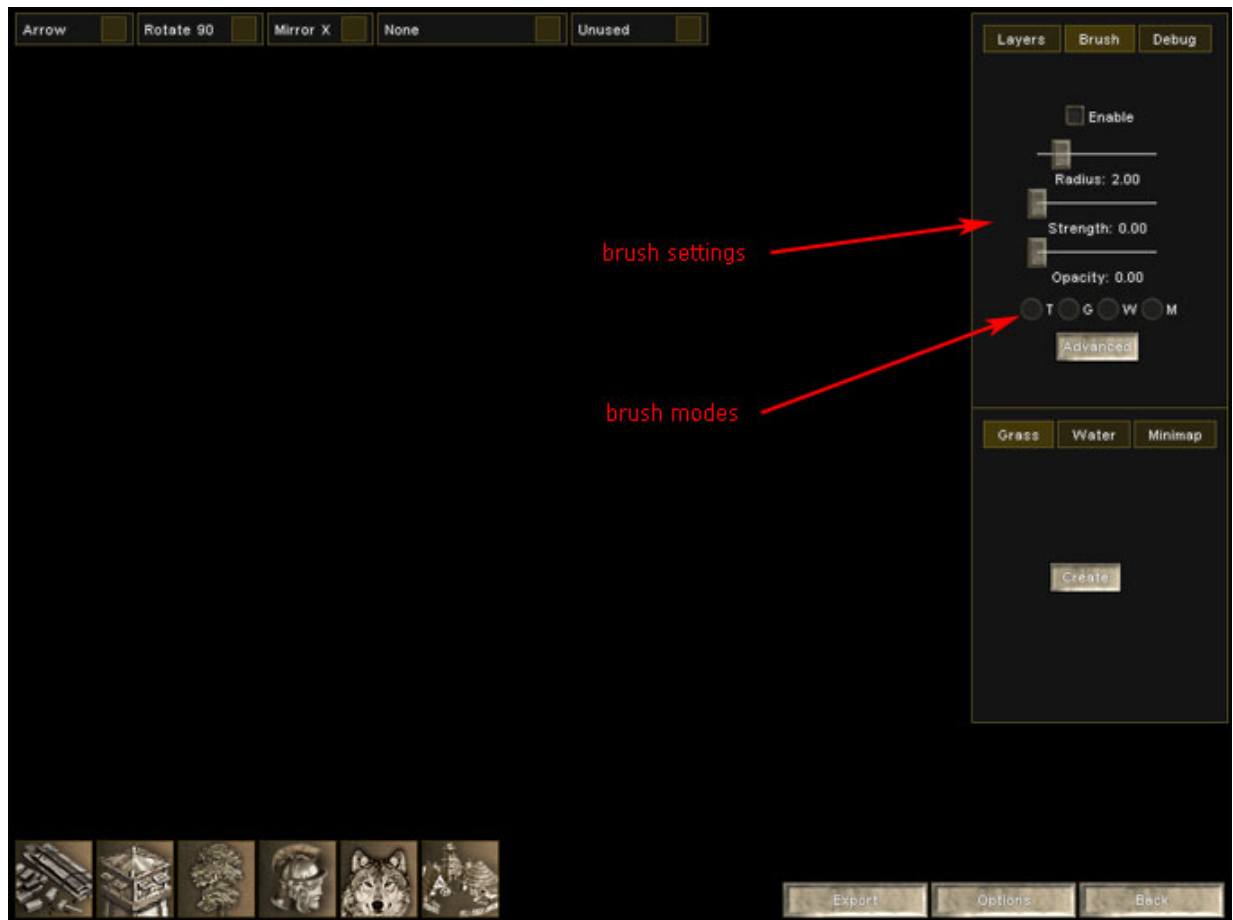


image 7

You can turn the brush on and off by pressing the "Enable" check box. You can set the radius of the brush, which determines how much of the terrain, around the brush's point-of-contact with the terrain, is affected. You can adjust the strength of the brush, which is used to determine how much you burn, heal, erase, restore, raise, and lower the terrain. And finally, you can set the opacity of the brush, which you set when using a marker or pastel brush type.

All of the brush types (*see image 6*), except arrow, apply only to the tile brush. There are so many things you can do to a tile like erase, burn, raise, etc. That is why there are so many types of tile brushes. So, basically, the toolbar at the top of the screen is used for tiles.

## Using layers

Once you have selected the brush, you will want to select the correct layer to paint on. You select a layer by pressing the circular radio button for that layer. Each brush mode is meant to be used with certain layers, i.e., a tile brush is used to paint on tile layers, a grass brush is used to paint on the grass layer, a water brush is used to paint on a water layer, and a model brush is used to paint on the nature, character, structure, village, and critter layers. You can hide an entire layer by pressing the check box for that layer. You can also delete everything on a layer by pressing the "X" button for that layer.

## Painting tiles

To "paint tiles" also means to place tiles on the terrain. When you start a new blank map, you will see only a grid since there are no tiles on the terrain. To begin painting tiles, you should start by enabling the brush, switching to tile brush mode, selecting the paint brush type, selecting a tile layer, and finally selecting a tile texture from the tile-set. It is helpful when you have the "spatial view", which shows the grid, enabled, which can be found under the "debug" tab.

You may want to rotate or mirror a tile texture. To do that, you select a rotation flag or a mirror flag and click any of the tiles in the 3D space. When you want to paint again, you switch back to the paint brush type.

## Adding tile logic flags

To start, enable the brush, switch to tile brush mode, select any tile layer, select tile flag brush type, and finally, select a tile flag. Remember, tiles can have many properties (i.e., logic flags). If you want to remove tile logic flags from any of the tiles, then press the space-bar while painting over the tiles that you want to remove the flag from.

## Adding tile types

Same as with tile logic flags, but instead, you select the tile logic brush type. Since a tile can be of only one type, you change this rather than remove it. In other words, the space-bar trick will not work.

## Painting grass

Start by creating a new grass area. To do that, press the "Create" button in the "grass tab". You can create a maximum of 12 grass areas. When you are ready, select the new grass area by pressing the circular radio button for that new area, enable the brush, switch to grass brush mode, select the tile logic brush type, select the grass layer, select the grass tile type, and finally, press the "draw logic" check box in the "debug" tab. LOL!

If you had made any mistakes while painting grass and are unsatisfied with the results, then you will have to delete the grass area and start over. You can delete a grass area by pressing the "X" button for that area.

## Drawing water

Start by creating a new water area. To do that, press the "create" button in the "water tab" (*see image 8*). Press the "add vertex" circular radio button and begin placing points on the terrain. While you are placing points, the editor will automatically try to connect them as best it can. If you had made any mistakes, just press the "cancel" button and start over. You can change the water's color and transparency by adjusting the sliders. Once you have finished drawing the water, you might see a few triangles that you want removed. You can remove them by pressing the "remove triangle" circular radio button and selecting those triangles. The final step is to let the editor know that you

are done with the water model and want it saved. You do that by pressing the "finalize" button. The editor will then modify the water model a bit by blending the edges and adding a material to it.

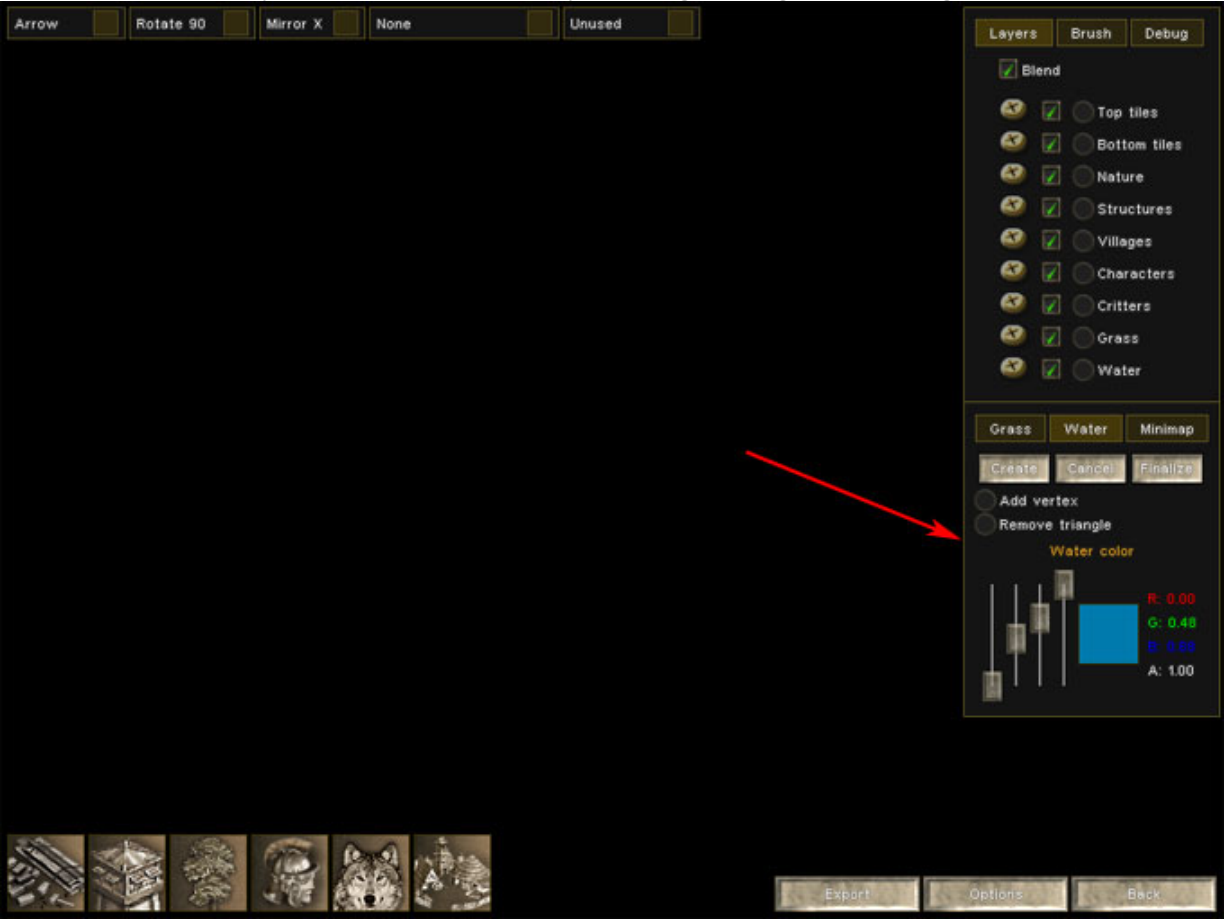


image 8

## Finishing your project

---

You may want to begin testing your map in the game towards the end of your development process. Once you are satisfied, you may want to convert it into a mod, package it, and then distribute it among other players. If you want to convert your map into a mod, then you will need the software necessary to package your map. Its always nice to have free stuff available. One free program is 7-Zip which can be found at <http://www.7-zip.org/>

## Testing your map

If you had chosen to export your map into the Praetorians "Data" folder then you are ready to test without touching any files. Otherwise, you can simply copy and paste both the "Mapas" and Misiones" folder into the Praetorians "Data" folder.

## Packaging your map

To package your map you add both the "Mapas" and "Misiones" folders to a ZIP file. You can rename the ZIP file using the extension PAK or PAT. A PAT file has a higher priority than a PAK.

## Common problems

---

### My chief cannot enter a village!

That is caused by the village's orientation. To correct the error, you will need to edit the MIS file. These are your options:

- adjust the value of *\*POSICION*
- adjust the value of *\*ORIENTACION* (radians)
- or do both

### I cannot build a garrison!

That is caused by the village's orientation. To correct the error, you will need to edit the MIS file. Offset the garrison from the village's center by adding this line:

- *\*PUNTO\_ANEXO* [x,y,z] (e.g. *\*PUNTO\_ANEXO* -6 0 3)

### Why is there is no sound?

Right. You will need to edit the MIS file. You can add sound at a certain position on the map by adding this:

```
*SONIDO
{
  *TIPO "APAJ"
  *POSICION [x,y,z] (e.g. *POSICION 19.71 2.01 33.50)
  *VOLUMEN 1
}
```

But how do I know where to position the sound? A trick I use is I save a copy of my current MIS file, which has critters and villages already, go back to the editor and export player positions, and then add the *\*SONIDO* code, using the player positions, to the MIS copy I had made earlier.

### Why is there is no rain, thunder, and lighting in the game!

You will need to activate the atmospheric effects. Add this:

- *\*EFECTO\_ATMOSFERICO*

=====

## Keyboard shortcuts

---

### General

C	preview 3D model before placing it on the map
H	hide graphical user interface
F8	capture screen-shot
F1	clear terrain alpha
F2	clear terrain color

### Camera movements

Q	down
E	up
W	forward
S	backward
A	left
D	right
right mouse button	rotate

## Conclusion

You have learned that installing the editor is as simple as unzipping the map editor into the Praetorians folder. You have seen that creating and editing projects is as simple as adding data to an XML document. You have learned about terrain and its different components. And you have learned how to both configure and use the editor.

There are also video tutorials available on the world wide web. The address is [http://www.youtube.com/profile\\_videos?user=kwantum26](http://www.youtube.com/profile_videos?user=kwantum26)

There is even a website that was setup by a Praetorians fan where you can upload and download new maps. The address is <http://praetoriansmaps.teamblind.de/>

If you think this manual sucks, then fuck you.

