

## Rules for property maps

Property maps are special voxel meshes which determine the material properties of a mesh by coloration. Coloring a voxel in a property map will change the property of the same voxel in the original mesh. For example, if you want the hilt of a sword to be solid and the blade opaque, you would color the hilt the solid color (255,255,255) and the blade transparent (128,128,128)

**Note – do not change the color of the origin voxel – it remains (255, 0, 255)**

There are three property maps – *type*, *alpha*(transparency amount), and *specular*(shininess)

1. Type map – this map determines whether a block is solid or transparent. Currently, there are three block types—Solid, glass and tiled glass
2. Alpha map – this map alters the transparency of any glass blocks.
3. Specular map – this map changes the specular property of any solid (non-glass) blocks.

Property maps use the following name convention:

<i>MeshName.qb</i>	Original Mesh
<i>MeshName_t.qb</i>	Type map
<i>MeshName_a.qb</i>	Alpha map
<i>MeshName_s.qb</i>	Specular map

### Colors for type map:

Solid(default)	(255, 255, 255)
Glass	(128, 128, 128)
Tiled Glass	(64, 64, 64)
Glowing Solid	(255, 0, 0)
Glowing Glass	(255,255,0)

### Colors for alpha map:

Very transparent	(16, 16, 16)
	(48, 48, 48)
	(80, 80, 80)
	(112, 112, 112)
	(144, 144, 144)
	(176, 176, 176)
	(208, 208, 208)
Opaque	(240, 240, 240)



### Colors for specular map:

Rough(default)	(128, 0, 0)
Metal	(0, 128, 0)
Water	(0, 0, 128)
Iridescent	(128, 128, 0)

### How to create a blueprint for testing

First, make sure your mesh is in the same directory as your maps. Next, using Windows Explorer, drag your mesh file over to **devtool\_convert\_to\_blueprint.bat** . This should process the file into a blueprint and save it in your Trove/blueprint directory. If the file wasn't created, check the log in the roaming directory (%appdata%/Trove/DevTool.log) to see if there was a problem loading the files.

### How to test a blueprint in-game

To test a weapon blueprint in-game, type ***/weaponpreview [nameofblueprint]*** or ***/wp [nameofblueprint]***, where [nameofblueprint] is the name of your blueprint file. For example, if you've created mace01.blueprint, you can type */wp mace01* to test it.