

Atlan Texture Modding

Page is a work in progress

Documentation on creating texture mods with Atlan. This information applies to both DOOM: The Dark Ages and DOOM: Eternal. For DOOM: Eternal, it's preferable to create texture mods using idStudio. If you can't use idStudio, this guide offers an alternative approach.

Getting Started

Creating texture mods requires the following resources:

1. **Atlan Resource Extractor** - Use this to extract material2 decls from the game. These plaintext files define the game's materials and directly reference image assets.
2. **Valen** - A resource extractor that can extract vanilla image files from the game. Atlan Resource Extractor does *NOT* support extracting textures. Use this instead.
3. **Atlan Mod Packager** - Packages your custom texture PNG files into a format the game recognizes. This is a necessary step to reduce mod loading times, as converting the images is an expensive process. You must use Atlan Mod Packager to prepare your finished texture mods for distribution
4. **Mod Config File** - When adding *new* images, it is *highly recommended* you use the aliasing system provided by your mod's configuration file! Image files have long names that don't clearly describe what piece of a mesh they're for! The aliasing system is helpful for organization, and prevents you from needing to change the extension on your PNGs to match the vanilla filename!

Supplemental resources may also exist for common areas of texture modding:

1. Dark Ages Slayer Skins: Check out Kuddly Kraken's **Creation Kit**

Workflow Overview

This is a basic overview of how to create a texture mod:

1. Assemble your modded textures using the techniques described below. Your finalized images must be PNGs.

2. When you're ready to publish your mod, use Atlan Mod Packager to package it into a zip file! This step is *required*. Atlan Mod Loader will not convert PNGs it finds in zip files. It expects zipped images to be converted in-advance.

While developing and testing your mod, remember to take advantage of Atlan Mod Loader's **unzipped mod loading capabilities**. This should reduce iteration time.

Replacing Existing Images

Replacing existing images is more convenient than adding original images to the game. This approach requires Valen v0.3.3 or higher (see above). Image replacement should be sufficient for the majority of texture mods.

First, use Valen to browse the game files for the images you want to replace. Export your images. You'll notice that the names of the exported images have the format: `name#<16-character-code>.png`

The `#<code>` is important! Atlan detects that code and uses it to determine the correct asset path for your image. This way you don't need to recreate the entire original image path, which is often very long and complicated!

In your mod's folder, your image should have the path `image/<...>#<code>.png`

The actual path or name before the `#code` does not matter. All that matters is the path start with `image/` and the name ends with `#<code>`. Remember: **do not modify the code**

```
// Some Examples: Both of these paths refer to the same image with code #1234567812345678
image/art/models/characters/doomslayer/torso/1001/doomslayer_torso_1011#1234567812345678.png
image/my_cool_slayer_mod#1234567812345678.png
```

Adding New Images

Adding original images to the game is more complicated than replacing existing textures. Please pay careful attention to all of the information in this section. If all you're doing is replacing existing textures, you can skip this section.

If you want to add an original image to the game, instead of replacing an existing image, you'll need to follow several more steps:

1. Reference your new images inside one (or multiple) material2 decls.
2. Determine the correct extension properties for your image

3. Specify your image's encoding information. This step is not always necessary depending on the image.

Material2 Decls

Once you've loaded your new image assets into the game, you must reference them in one or more material2 decls! You can edit a vanilla decl to use your texture, or create an entirely new material2 to reference in other files. As the mod author, you make the design decisions!

Extension Properties

When deciding the asset name of your new image, you can choose any unique name:

```
// Remember: The modloader converts '@' symbols into '/' symbols
image/my_awesome_image.png
image@your_username_here@your_modname_here@image_1.png
image/cool_skin/helmet/helmet_albedo.png
```

However, the extension properties **must be written correctly**. If you fail to write them accurately, the game will not be able to find and load your modded image!

The exact type varies by the type of image you're adding, and by the inherited template of the material2 decl you're using it in. For example, if you're adding an albedo image, your extension properties should be `$streamed$mtlkind=albedo`

There is no exhaustive list of correct extension properties. The best way to be certain you're correct, is by studying the names of vanilla images produced by that particular material2 template. More obscure templates may require some trial and error to get the extension properties correct.

Revision #2

Created 22 October 2026 14:31:30 by FlavorfulGecko5

Updated 22 October 2026 16:25:07 by FlavorfulGecko5