Drakkar Cybernetics Documentation

- Intro
- Multimaterial Trim Shader
- Using the shader via script
- Custom Vinyl Decals
- Import Settings
- Links

Drakkar Cybernetics



The Drakkar Cybernetics series provides detailed robotic models designed for both cinematic and real-time applications, these models offer exceptional quality without compromising performance.

All characters feature **modular parts**, allowing you to mix and match arms, legs, torsos, and heads to create unique variations. The package includes **many base skins** and **customizable vinyl decals** to further personalize their appearance.

All models come in two fashions: **single mesh** and with head,arms,torso and legs as **separate meshes**.

A **custom shader** provides extensive material customization, from simple color adjustments to advanced effects like **carbon fiber textures and depth-enhancing details**. Two shader versions are available—one optimized for performance and another featuring **Parallax Occlusion Mapping** for an enhanced visual experience.

Multimaterial Trim Shader

The shader comes in two fashions:

- **MultiMaterial Trim**: designed for the best visual quality, including the Parallax Occlusion Mapping.
- **MultiMaterial Trim Simple**: designed for performance.

Please note that the advanced version still has some features that allows you to scale its computational weight based on distance (see later in this document).

The shader internally handles **up to four sub-materials** and **three glowing colors**.
Please note that they don't represent Unity's materials but they are embedded all in **one Unity material** to save draw calls and performance.

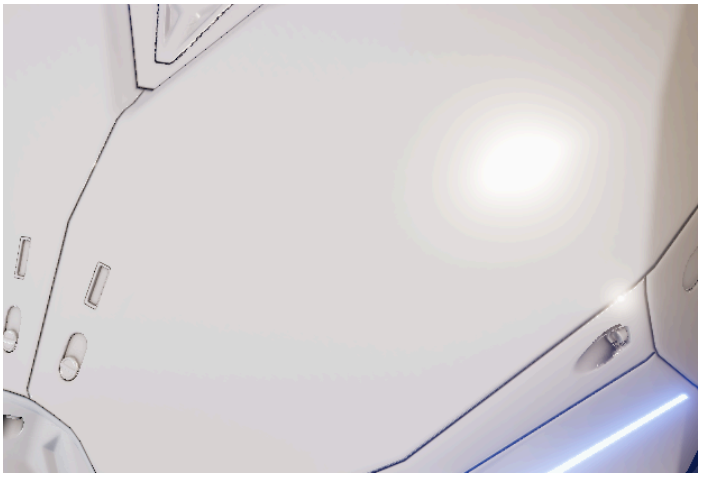
Parameters:

- **Surface Texture:** defines the texture that will handle the material (plastic/rubber/metal/etc) features the shader will use to represent a material. It should be set with the provided texture: **HighTeckFabrick_packed**.
- **Tiling**: defines how much tiled the surface texture should appear. It strongly depends on the size of the model. For a humanoid size the value **1.5** is a good choice.
- **Blend**: defines the offset the surface texture should have.
- **Trim Sheet**: is the texture defining all the mechanical details. It defaults to **Greebles_channels**.
- **Normal**: is the normal texture of the Trim Sheet. Default: **Greebles_Normal**.
- **Vinyl Select**: selects which vinyl set the model uses. A value of zero means no vinyls.
- **Vinyl Material**: defines which one of the sub-materials the vinyls will use.
- **Vinyl Trim Sheet**: The texture defining the vinyl decals.
- **Material Definition**
    - **Color 0-1-2-3**: define the main color of the four materials the shader handles.
        - **The Alpha channel** represents the smoothness value *applied to the surface texture*. It means a high alpha value will result in a shiny surface with many details on it like the one in the picture.



    - **Emissive 1-2-3**: three HDR colors that can be used by the model.
    - **Metalness**: define the metalness of each of the four sub-materials.

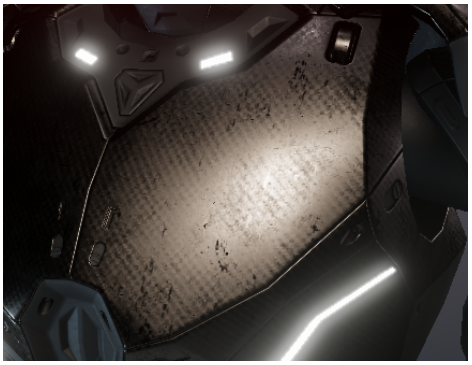
- **Smoothness Offset**: define how much smoothness (for each sub-material) should be **added** to the one defined in the alpha channel of the **Colors**. This smoothness value does not apply to the surface texture resulting in a perfectly smooth material with no details on it (see the picture).



Combining this value with the Colors' alpha channels you can obtain all degrees of smoothness and detail you like.

- **Pattern**: defines which pattern each sub-material will use.
  Patterns are color variations that can reproduce different kinds of special materials like carbon fiber.
  There are **two main patterns**.
  A value of zero means the **first pattern**, a value of **one** means the **second pattern**. Any value between zero and one is not recommended, it could lead to bad results.

- **Pattern Intensity**: defines how much of the pattern shows on each one of the four sub-materials. For instance, a value of 0.5 can lead to the result shown in the picture.
  Please note that **negative values** are allowed too. They just look like the inverted result of the positive ones.



Pattern influences the material smoothness! Keep it in mind when using patterns.

- **Color Offset**: controls, for each one of the four materials, the amount of "dirt" on them.
  A value of **zero** represents a **quite clean** material, a value of **one** is a **perfectly clean** material, any **negative value** leads to **dirty/very dirty** materials.

- **Parallax Occlusion Mapping** *(available only with the advanced shader)*

    - **Parallax Crop Distance**: defines the distance (in meters) from the camera where the parallax effect will be clamped upon other surfaces. It is a form of shader optimization.

    - **Angle Tolerance**: define the steepness of the angle where the parallax effect stops working to avoid details being “cut” by the real geometry.

    - **Parallax Max/Min Distance**: define the max and minimum distances where the parallax effect should take place.
      The **Min** value represents the distance where the parallax effect shows its best quality *(full effect)*.
      The **Max** value represents the distance where the parallax effect shows its lowest quality *(no effect)*.

    - **Reference Plane**: represents the gray level that is considered “flat” by the parallax effect. Default value: **0.5**.

    - **Quality Multipliers**: the first 3 values (the fourth is not used at the moment) defines the quality of the parallax effect in this order:
      1) the minimum samples *(default 8)*.
      2) the maximum samples *(default 16)*.
      3) the sidewall steps *(default 4)*.

      A value of 1 represents the default value (it means each of the values is multiplied by 1). Higher than 1 values increases the quality, lower than 1 values decreases it.

Using the shader via script

It is possible to tweak the shader values via script in real-time by accessing the global shader variable: "**ParallaxOffsets"**. It is a good way to tune the performances of your game based on different hardware/platforms.

```
Shader.SetGlobalVector("ParallaxOffsets",myVector);
```

Its x,y,z components just add to the parallax values:

X: adds to the *minimum samples*.
Y: adds to the *maximum samples*.
Z: adds to the *sidewall steps*.

The default values for minimum samples, maximum samples and sidewall steps are: **8**,**16**,**4**.

if we set the **ParallaxOffsets** to 5,3,4…

```
Shader.SetGlobalVector("ParallaxOffsets",new Vector4(5,3,4,0));
```

We will get:

minimum samples: 13 *(8+5)*
maximum samples: 19 *(16+3)*
sidewall steps: 8 *(4+4)*

Resetting **ParallaxOffsets** to 0,0,0,0 will restore the default quality values.

```
Shader.SetGlobalVector("ParallaxOffsets",Vector4.zero);
```

Custom Vinyl Decals

Drakkar Cybernetics comes with 4 sets of vinyl decals. They are stored in the texture: *Drakkar/Cybernetics/Textures/**Vinyls.tif***.

It is made of 4 channels: **R**,**G**,**B**,**A** each one contains a set of decals.

Please note that the decals at the bottom of the texture stay the same in each set. This is just a design choice and it's not a strict rule.











If you open the texture in your favourite image editor (Photoshop, Krita, etc.) **you can modify each decal at will**.
To be sure everything won't break you should be aware of the following **guidelines**:
Import the image:
*Drakkar/Cybernetics/Drakkar Hardsurface Commons/Vinyls Guidelines/**Vinyls_Guides.png***
as a layer on top of the texture. It will look like this:



The **red lines** should never be crossed. They are the boundaries of each decal.
The **blue areas** are safe margins you should keep free from graphics to allow the decals to not interfere with each other.

You can modify each channel or create new standalone images and pack them into **RGBA channels** to use them as vinyl decals with **Drakkar Cybernetics** models.

Remember to **remove or hide the Guides layer** before exporting the final texture.

Import Settings

All models of Drakkar Cybernetics must be imported with the following settings:

Models

Normals: Import
Normals Mode: Unweighted
Smoothness Sources: From Smoothing Groups
Tangents: Calculate: Mikktspace

Generate Lightmap UVs: OFF *(all models use UV1,UV2,UV3 for shader reasons)*

Skin Weights: Custom
Max Bones/vertex: 1 *(except when specified)*

Rig: Generic or Humanoid *(the sample assets are set to Humanoid)*

Textures

All textures Compression should be set to “High Quality”

Links

DRAKKARDEV.COM
EMAIL
DISCORD
FACEBOOK
YOUTUBE
TWITTER