[1.7.x] Textures Unlimited - PBR-Shader, Texture Set, and Model Loading API

Follow 83

By Shadowmage, November 11, 2017 in Add-on Releases

• plugin • noparts • shaders • physicallybasedrendering

Rate this topic
★★★★★

Start new topic Reply to this topic

1 2 3 4 5 6 NEXT » Page 1 of 56 ▾

Shadowmage
Sr. Spacecraft Engineer
●●●●●

Members
+ 6,304
4,341 posts

Posted November 11, 2017 (edited) Report post

Textures Unlimited

The Textures Unlimited addon brings the Unity 5 Standard shader functions to KSP along with a suite of utilities for loading shader and model asset bundles, and a host of shader and texture-related functions to support them. It does not add any new part or textures by itself, but is intended to be used and distributed by other mods who do not wish to develop their own plugin code or shaders.

Features:

- Asset-Bundle Shader pack loading system
- Reflection Probe Manager - updates the reflection probes in a fashion that works in KSP, with quality and performance configuration available
  - Reflective textures are fully supported through the PBR shaders
- Texture Replacement system - replace the textures

on any .mu model, in a mesh-by-mesh fashion; complex material arrangements fully supported

- TextureSwitch module - includes a simple texture-switch module that can be added to parts through MM patches to facilitate in-game user-driven texture switching
- User-driven 'recoloring' system for parts. Define the pattern on the part through a mask texture, and let the user specify the color scheme. Legacy and PBR versions available. Requires special texture and shader setup.
- Asset-Bundle Model loading system -- supports loading of BlendShape/ShapKey models compiled as AssetBundles (and/or any other model). Supports loading of models using non-KSP shaders.
- List of included shaders; all are compatible with KSPs part-highlighting and thermal glow functions:
  - SSTU/PBR/Metallic -- Unity 5 Standard-Metallic workflow, diff/met/nrm/ao/glow
  - SSTU/PBR/MetallicTransparent - Same as above, but with transparency/opacity support.
  - SSTU/PBR/StockMetallicSpecular -- uses stock Diff-spec(a) texture, with user-provided MET maps. Aimed at converting stock parts to PBR rendering.
  - SSTU/PBR/Masked -- PBR-Metallic based 'recolorable' texture system
  - SSTU/PBR/Solar - Subsurface-Scattering based back-lit solar panel shader. Uses a GLOW/thickness map to determine backlight effect, along with several tunable shader parameters.
  - SSTU/Masked -- Legacy Diff/Spec based 'recolorable' texture system
  - SSTU/SolarShader -- Legacy Diff/Spec solar-panel shader. Includes a GLOW map for a view-direction dependent back-lit effect.
- Scatterer / EVE / SVE fully supported.
- Any/all of these functions are optional. Use only the features that you want or need.

Potential uses

- Mods wanting to use PBR textures and rendering for their parts

- Mods needing custom shaders for their parts
- Mods needing BlendShape/ShapeKey animated models
- PBR stock-conversion mods

Downloads:

The latest releases will always be available from:

https://github.com/shadowmage45/TexturesUnlimited/releases

Dependencies:

Requires Module Manager (not yet included)

User-created Configs and Texture Packs:

(PM me to be added to the list, not required for use):

Used in the following mods:

(PM me to be added to the list, not required for use):

Legal/Licensing

Licensed under GPL 3.0 or later (https://github.com/shadowmage45/TexturesUnlimited/blob/master/LICENSE). Source for the plugin and shaders are provided online at:

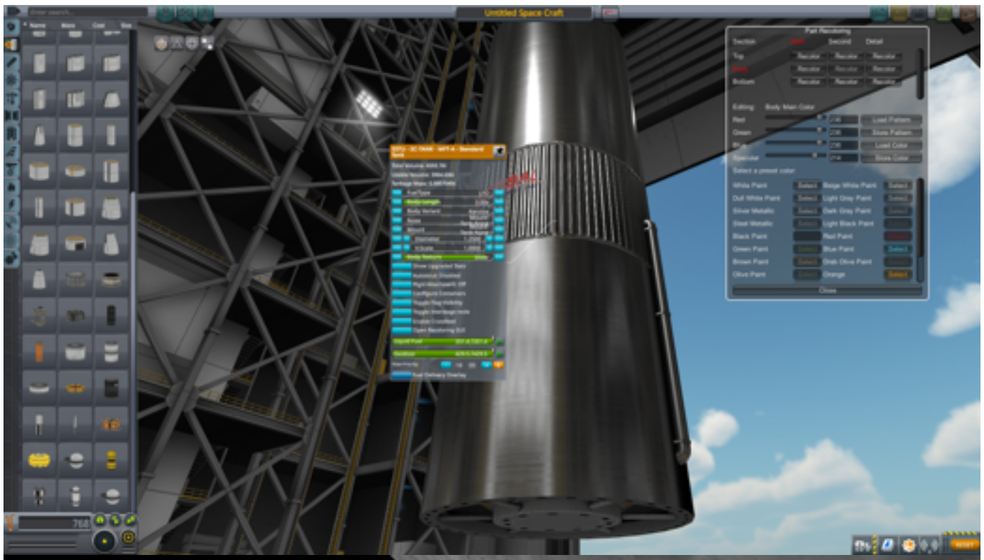
(https://github.com/shadowmage45/TexturesUnlimited)

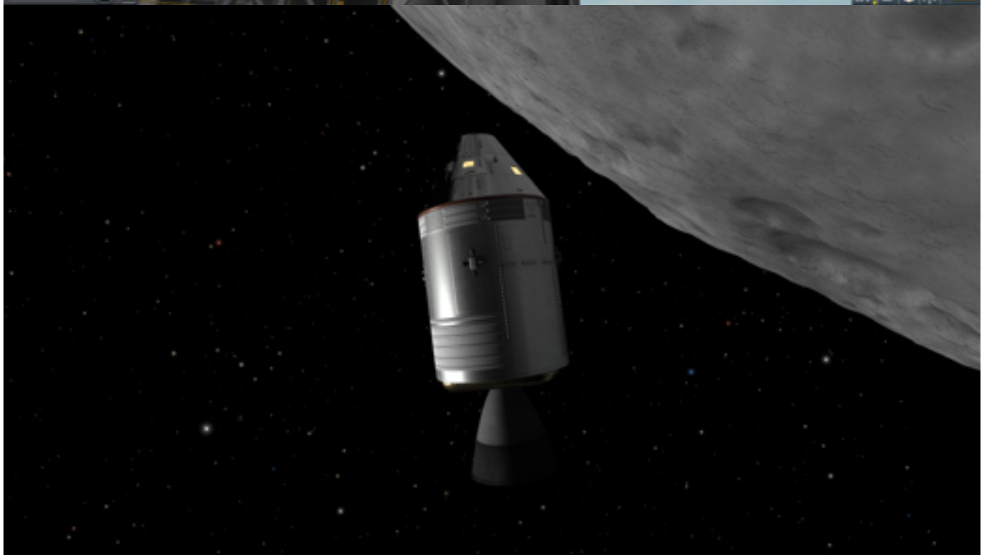
You are free to use and distribute this mod as long as the terms of the license are followed. Generally speaking this should place no restrictions if an unmodified copy is bundled in full and included in third party mod's releases. Modified copies are subject to further restrictions as are detailed in the license.

Examples

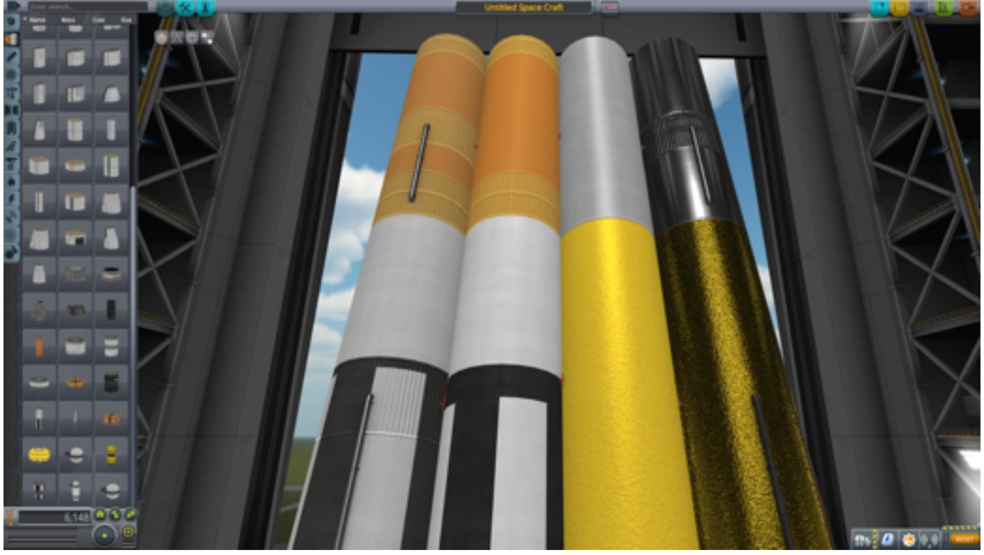
Examples of some of the capabilities of the shaders (all examples using parts from SSTU)

PBR - Reflective and semi-reflective (reflections can range from mirror-sharp to dull based on texture setup)

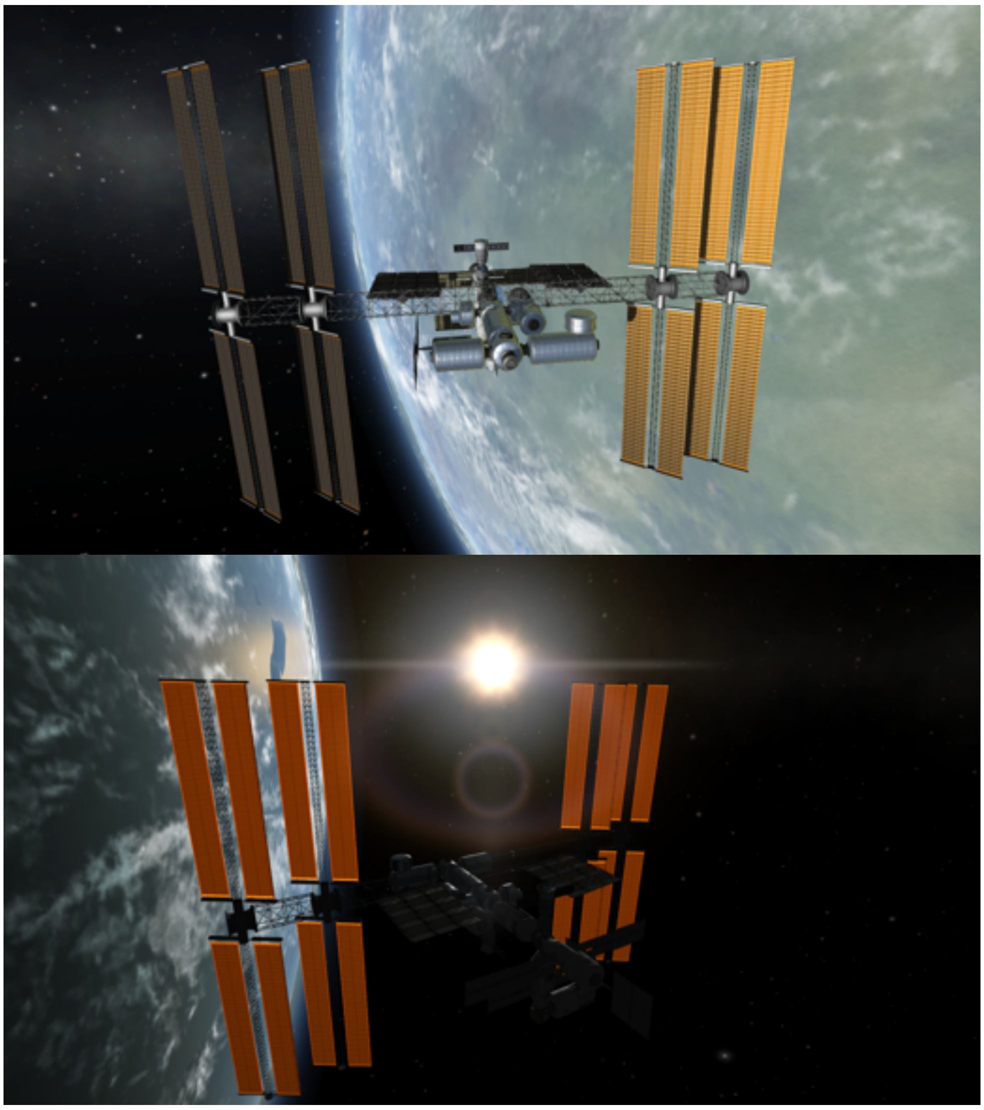




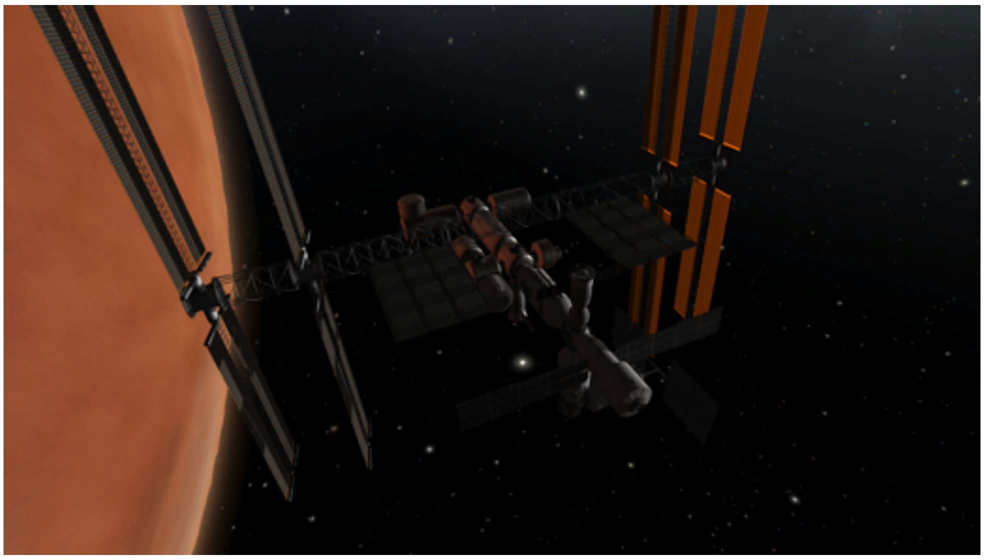
Masked/Recoloring (PBR and legacy versions, require special texture setups)

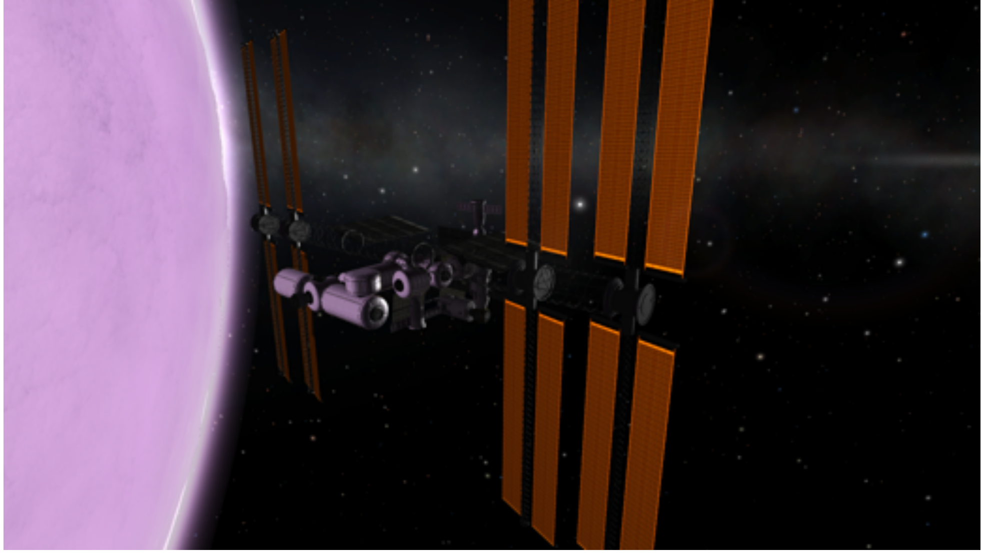


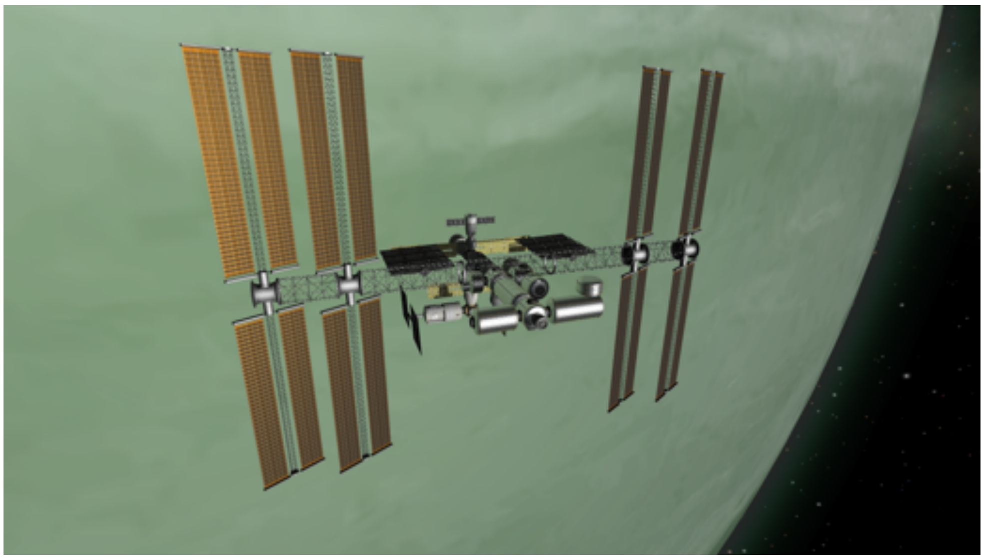
Back-lit Solar Panel Shader + PBR shaders (+EVE + Scatterer) (solar panels still using legacy backlit shader)



Reflection of surroundings; colored ambient light handling







Technical Support

- All requests for support must be done through the GitHub issues page: https://github.com/shadowmage45/TexturesUnlimited/issues
- All issue tickets must be accompanied by a KSP.log file

Known Issues

- DX9 Has cube-map blurring problems with PBR shaders, results in sharp lines in reflections even on non-reflective surfaces -- use OpenGL
    - This is a Unity Engine problem, and not one that can be solved or worked around by plugin code.
    - Unity does not expose the hardware based convolution functions that would be needed (and it may be those functions that have

issues with DX9).

- DX9 Has incorrect faces in cube-maps, resulting in very poor reflections.
- Kerbals and other vessels are not visible in reflections.
- Momentary lack of reflections upon scene load.
- Scatterer not present in reflections.
- Any other issues listed on the issues tracker: https://github.com/shadowmage45/TexturesUnlimited/issues

Future Plans

- Documentation on shaders properties and texture setup
- Documentation on configs for shader and texture replacement and texture switch module
- Shader variant integration -- use keywords to drive shader features. Don't sample/process features that aren't needed (more optimal rendering).
- More comprehensive texture-replacement methods; skyboxes, kerbal suits, ksc buildings?
- Possible inclusion of post-processing support, at least bloom
- Investigation of HDR rendering support in KSP

Credits / Contributors

- @Lilleman for his original work on using PBR shaders in KSP. While I didn't use any of his code directly, I did learn a few things from it.
- @HaArLiNsH for his updated/maintained TextureReplaceReplaced, which provided valuable insight into the rendering setup used by KSP
- @shaw for his work on TextureReplacer (legacy), and @rbray89 for his original work on the reflection code.
- @blackrack for including workarounds in Scatterer that allow for integration with TexturesUnlimited (by using the TR reflection camera naming scheme).
- @Electrocutor for his work on the Stock conversion

patches, and for helping track down issues with the code/configs/shaders.

Edited June 22 by Shadowmage

+ Quote

SamBelanger, Bonus Eventus, Electrocutor and 64 others like this

SSTU
Shadow Space Technologies Unlimited

SS
Low Part Co
Lifters, Orbiters,
Modular Parts an
(Deve

tater

Rocket Surgeon

●●●●●

Members

18,204

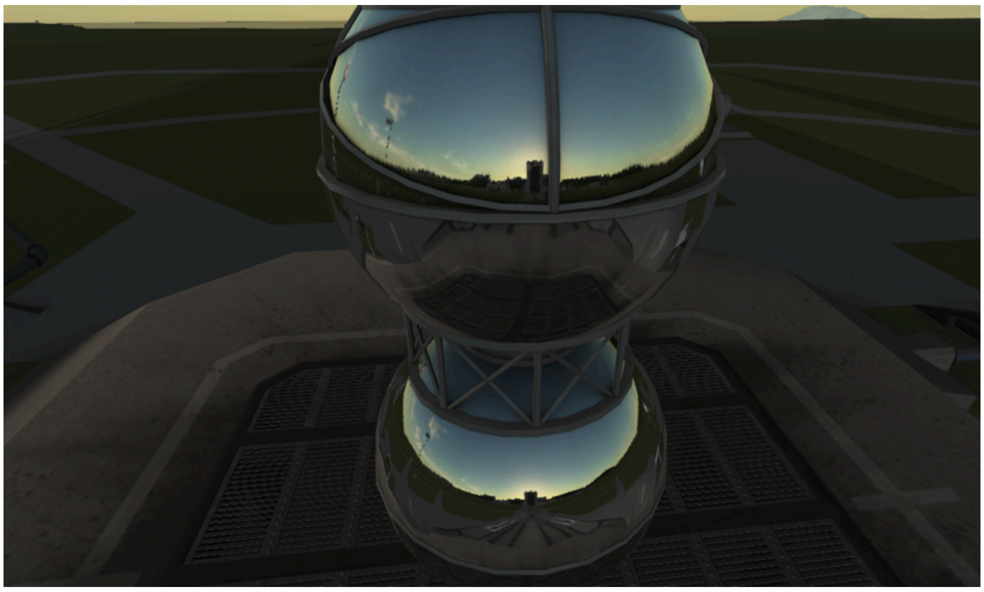
13,683 posts

Location: On the side of a mountain in New Mexico

Posted November 11, 2017 (edited)

The effect in game is really stunning.

Notice even the EVE clouds reflected:



(on my 7.5 year old computer, BTW).

Edited November 11, 2017 by tater

+ Quote

DeltaDizzy, lajoswinkler, TheProtagonists and 22 others like this

Awesome SSTU stuff   Forum Launch Coverage Index

bcink

JWST Kerbal Engineer

●●●

Members

1,377

438 posts

Posted November 11, 2017

Report post

> **On 11/11/2017 at 6:23 PM, tater said:**
>
> The effect in game is really stunning.
>
> Notice even the EVE clouds reflected:
>
> (on my 7.5 year old computer, BTW).

Wow! That's nice...

Quote

tater, AndrewDraws and Shadowmage like this

James Webb [WIP] Thread Kerbalow Aerospace [WIP] Thread DSSHU [WIP] Thread Making Alternate History [WIP] Thread Duna Direct [WIP] Thread

Electrocutor

Tinkerer

●●●●

Members

600

617 posts

Posted November 11, 2017 (edited)

Report post

Do the included sub-shaders also include the ability to define a metallic mask texture? (To be able to properly set it for existing non-PBR stock models)

> **Quote**
>
> With a texture assigned to the Metallic parameter, both the Metallic and Smoothness sliders will disappear. Instead, the Metallic levels for the material are controlled by the values in the Red channel of the texture, and the Smoothness levels for the material are controlled by the Alpha channel of the texture.

Edited November 11, 2017 by Electrocutor

+ Quote Shadowmage likes this

Shadowmage

Sr. Spacecraft Engineer

●●●●●

Members

⊕ 6,304

4,341 posts

Posted November 11, 2017 Report post

> **On 11/11/2017 at 8:07 PM, Electrocutor said:**
>
> Do the included sub-shaders also include the ability to define a metallic mask texture? (To be able to properly set it for existing non-PBR stock models)

Yes, there is a shader specifically geared towards using an existing stock KSP/Specular or KSP/BumpedSpecular (and emissive variants) texture setup, that accepts a grayscale MET mask. It reads base color from diffuse RGB, gloss from diffuse A, and metal from metal mask R. That is the SSTU/PBR/StockMetallicSpecular shader. It differs from the main PBR shader only from what textures/channels it reads the gloss from.

For stock parts using the KSP/Diffuse or KSP/BumpedDiffuse, you would use the 'main' SSTU/PBR/Metallic shader. PBR shading can't work properly without some sort of gloss and metallic inputs, so these are supplied in the same format as the Unity 5 Standard shader -- metal in RGB, gloss in A.

+ Quote

SS

Low Part Co

Lifters, Orbiters,

Modular Parts an

(Deve

Electrocutor

Tinkerer

●●●●

Posted November 11, 2017 Report post

I declare yourself a godsend, sir.

Members
⊕ 600
617 posts

> **Quote**
>
> - Kerbals and other vessels are not visible in reflections.

You probably just need a different size or mask on your reflection probe. https://wiki.kerbalspaceprogram.com/wiki/API:Layers

+ Quote

Space Scumbag

Sr. Spacecraft Engineer
●●●

Members
⊕ 387
331 posts

Posted November 11, 2017 Report post

Wow, that is awesome. Thanks for creating this mod!

+ Quote bcink likes this

Electrocutor

Tinkerer
●●●●

Members
⊕ 600
617 posts

Posted November 12, 2017 Report post

Could you please add to the original post, within spoiler blocks:

- An example of the simplest .cfg usage for each functionality
- An example of the most complicated usage for each functionality
- A section that describes each Shader by name, what effects it includes, all properties that can be set for it, and the specifics of how those properties should be formatted (i.e. what each color channel does and whether or not the alpha channel is used, also stating whether the DDS should be in DX1, DX5, or

DX5nm).

- Include a link to Object Inspector: https://github.com/linuxgurugamer/KerbalObjectInspector

+ Quote Shadowmage likes this

riocrokite

Lego Rocketeer

●●●●

Members

608

855 posts

Posted November 12, 2017 Report post

this is great Shadow! have an upvote:)

+ Quote Shadowmage and Jimbodiah like this

Kermangeddon Industries

Old mods: Inline Ballutes [IB], Mobile Frame System [MFS], Stork Delivery System [SDS]

Current mod: [SAFIRE] Stockalike Adjustments to Fuels, ISRU, Ratios and Engines

Shadowmage

Sr. Spacecraft Engineer

●●●●●

Members

6,304

4,341 posts

Posted November 12, 2017 (edited) Report post

> **On 11/12/2017 at 12:54 PM, Electrocutor said:**
>
> Could you please add to the original post, within spoiler blocks:
>
> - An example of the simplest .cfg usage for each functionality
> - An example of the most complicated usage for each functionality
> - A section that describes each Shader by name, what effects it includes, all properties that can be set for it, and the specifics of how those properties should be formatted (i.e. what each color channel does and whether or not the alpha channel is used, also stating whether the DDS should be in DX1, DX5, or DX5nm).
> - Include a link to Object Inspector: https://github.com/linuxgurugamer/

> KerbalObjectInspector

All good suggestions. Apologies for not having some of that ready at time of release... I was specifically pushing the release out earlier so that @bcink could release the James Webb mod.

For existing documentation on the texture switch / default texture replacement -- please see the SSTULabs repository. I'm using the texture switch stuff all over the place. https://github.com/shadowmage45/SSTULabs/blob/recolorExpansion/GameData/SSTU/Data/TextureSets/TextureSets-SC-A - Soyuz.cfg. Default shader-assignments (use non-KSP shaders, without using texture-switch module; operates directly on .mu models) https://github.com/shadowmage45/SSTULabs/blob/recolorExpansion/GameData/SSTU/Shaders/DefaultShaderAssignments-SolarPanels.cfg (Please keep in mind there is no 'most complicated' limit for the texture sets; you can make them as long and complicated as the model setup requires)

Shader properties -- Will work on getting something written up on those throughout this week.

As far as the DDS specs go... I'm not here to teach people graphics; if they don't already know the proper DXT format for what they are doing, they have far bigger concerns than I can solve.

Edit: Should add that the DDS discussion has nothing to do with the shaders, and is entirely depenent on the content being stored in the texture (the shaders will accept whatever texture data is uploaded to them regardless of its file format; png, tga, mbm, dds, anything KSP will load).

- DXT5nm = all normal maps
- DXT5 = RGBA textures; anything with an alpha channel (and grayscale textures with alpha; e.g. MET/GLOSS maps)
- DXT1 = RGB - no alpha channel (and grayscale textures with no alpha; e.g. AO maps)

Edited November 12, 2017 by Shadowmage

+ Quote tater and bcink like this

SS
Low Part Co
Lifters, Orbiters,
Modular Parts an
(Deve

Electrocutor

Tinkerer

●●●●

Members

+ 600

617 posts

Posted November 12, 2017

Report post

> **Quote**
>
> (Please keep in mind there is no 'most complicated' limit for the texture sets; you can make them as long and complicated as the model setup requires)

For the 'most complicated' example, I just meant to include all possible cfg variables. A simple example may not need to specify anything except model, shader, and _BumpMap if it's just an example of how to add BumpMap support to an existing stock model.

Asking this made me curious though, do you default texture values to the already existing values? For example, if you just wanted to add BumpMap to a Mk1-2 Command Pod, would you need to re-specify the diffuse and emission maps, or could you just add the _BumpMap and the others would pull in by default?

+ Quote Shadowmage likes this

Electrocutor

Tinkerer

●●●●

Posted November 12, 2017

Report post

I've been trying to get a simple example to work, but not having good luck. Do see what might be wrong?

Members

⊕ 600

617 posts

```
KSP_Model_Shader
{
        name = Glass
        model =
Squad/Parts/Command/Mk1-2Pod/model
        TEXTURE
        {
                shader =
SSTU/SolarShader
                mesh = SideWindow
                mesh = FrontWindow

                PROPERTY
                {
                        name =
_Opacity
                        float = 0.5
                }
```

+ Quote

Duhya likes this

Shadowmage

Sr. Spacecraft Engineer

●●●●●

Members

⊕ 6,304

4,341 posts

Posted November 12, 2017 (edited)

Report post

> **On 11/12/2017 at 4:10 PM, Electrocutor said:**
>
> Asking this made me curious though, do you default texture values to the already existing values? For example, if you just wanted to add BumpMap to a Mk1-2 Command Pod, would you need to re-specify the diffuse and emission maps, or could you just add the _BumpMap and the others would pull in by default?

It should use existing values for anything already present.

What happens, from a technical perspective, is:

- Plugin finds the model and specific mesh(es) to adjust
- Grabs a reference to the existing material, makes it

unique for those meshes (so as to not inadvertently adjust the entire model)
- Updates the shader, if specified (if you don't specify a shader, it won't touch it)
- Adjusts any textures and/or properties specified. Anything without a line in the config does not get touched.

> **On 11/12/2017 at 4:40 PM, Electrocutor said:**
>
> I've been trying to get a simple example to work, but not having good luck. Do see what might be wrong?

The first line needs to be all capital letters:

KSP_MODEL_SHADER

The SolarShader also expects specific _SpecMap and _Emissive texture inputs, and might not work properly for command-pod windows -- the back-lighting is view-and-light-directionally-dependent, simulating partially polarized light diffusing through a transluscent object. There are a few other properties it uses to determine the back-lit effect, examples of which can be seen: https://github.com/shadowmage45/SSTULabs/blob/recolorExpansion/GameData/SSTU/Shaders/DefaultShaderAssignments-SolarPanels.cfg

I should also add that any of the .shader files can be placed directly into the Unity Editor, and you can play around with the texture slots and property values there to see exactly what some of them do.

Edit: these files here: https://github.com/shadowmage45/TexturesUnlimited/tree/master/CustomShaders

Grab them all and drop them in your Unity editor project Assets folder somewhere; they will become available as shader options for materials. Note that KSP-PartTools does not support loading of models exported with those shaders, but it can let you know what to expect from their use in-game. Some of they may or may not be fully functional yet; I'm still working on cleaning up a few and adding in support for the Unity-5 Standard Specular workflow as well.

Edited November 12, 2017 by Shadowmage

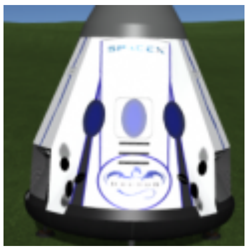
+ Quote HebaruSan likes this

SS
Low Part Co
Lifters, Orbiters,
Modular Parts an
(Deve

harrisjosh2711

Dragon Rider

●●●●



Members

+ 512

796 posts

Location: Kermerica

Posted November 12, 2017 Report post

My vote for modder of the year award.....

+ Quote Mecripp likes this

Electrocutor

Tinkerer

●●●●

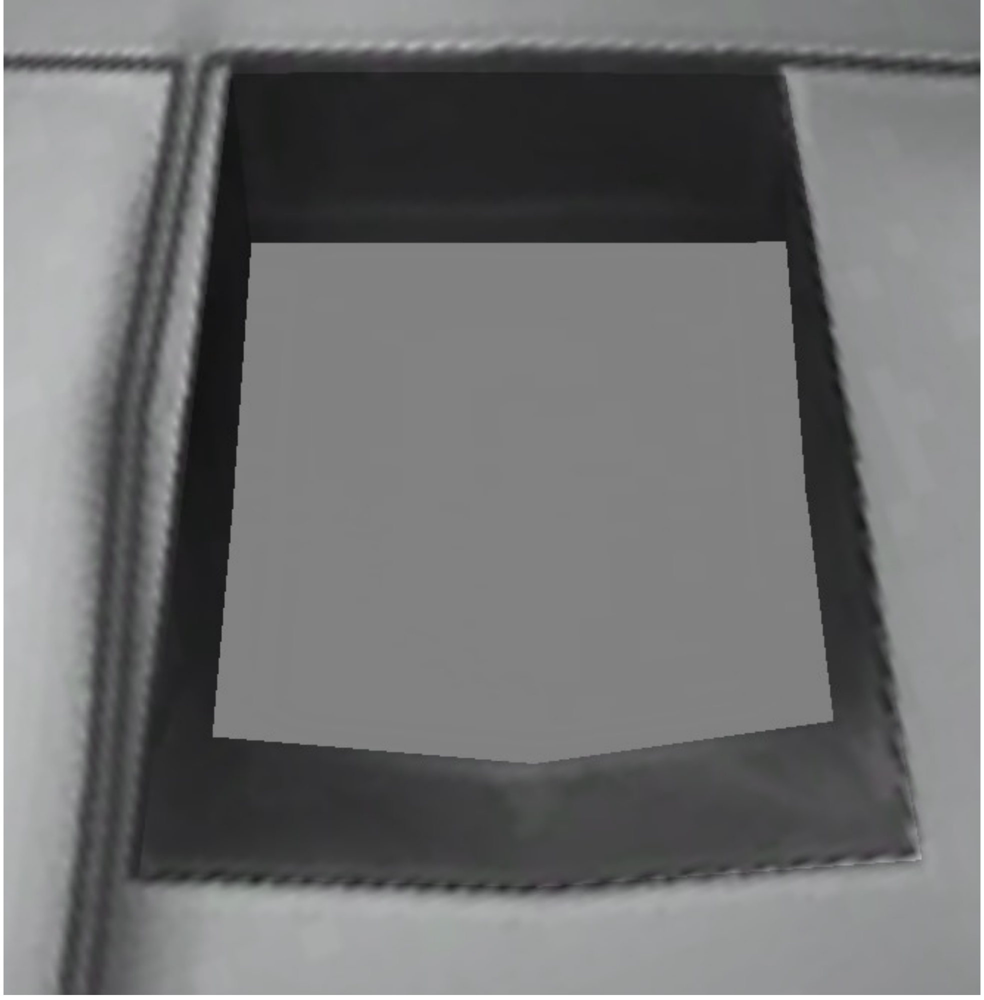
Members

+ 600

617 posts

Posted November 12, 2017 Report post

Thank you for the pointers. I'm still unable to get any reflections though. Plus, setting the texture to 100% black ends up making it gray?



```
KSP_MODEL_SHADER
{
        name = Glass
        model =
Squad/Parts/Command/Mk1-2Pod/model
        TEXTURE
        {
                shader =
SSTU/PBR/StockMetallicBumped
                mesh = SideWindow
                mesh = FrontWindow

                texture =
_MainTex,black
        }
}
```

+ Quote

Shadowmage

Sr. Spacecraft Engineer

●●●●●

Members

6,304

4,341 posts

Posted November 12, 2017

Report post

> **On 11/12/2017 at 6:26 PM, Electrocutor said:**
>
> Thank you for the pointers. I'm still unable to get any reflections though. Plus, setting the texture to 100% black ends up making it gray?

In order to get reflections, you will need to enable the reflections in config; a simple patch to do so is:

```
@REFLECTION_CONFIG[default]
{
        %enabled = true
}
```

As far as the config goes, the texture definition needs to include the path of the texture relative to GameData folder. You might also need to supply a metallic map for the shader to work properly. An example might be, if your textures were located in GameData/PBRTest/Assets/<texture files>:

```
KSP_MODEL_SHADER
{
        name = Glass
        model =
Squad/Parts/Command/Mk1-2Pod/model
        TEXTURE
        {
                shader =
SSTU/PBR/StockMetallicBumped
                mesh = SideWindow
                mesh = FrontWindow
                texture = _MainTex,
PBRTest/Assets/black
                texture =
_MetallicGlossMap,
PBRTest/Assets/metalMask
        }
}
```

(they gray you are seeing is likely because it was not finding the texture)

+ Quote

SS
Low Part Co
Lifters, Orbiters,
Modular Parts an
(Deve

Electrocutor

Tinkerer

●●●●

Members

Posted November 12, 2017 (edited)

Report post

"white", "black", "red", and "bump" are internal unity keyword textures. (1,1,1,1) (0,0,0,0) (1,0,0,0) (0.5,0.5,1,0.5)

but it seems like you are actually looking for a file named "black" or "white" and if not, don't set it?

⊕ 600
617 posts

Thanks for helping with turning reflections on; I had thought they were enabled by default.

Edited November 12, 2017 by Electrocutor

+ Quote

Agustin

Junior Rocket Scientist
●●●●

Members
⊕ 120
525 posts

Posted November 12, 2017 Report post

I have to do something to change direct9 to dx11 or opengl? Although textures did go wrong last timme I touched something about that....

+ Quote

Electrocutor

Tinkerer
●●●●

Members
⊕ 600
617 posts

Posted November 12, 2017 Report post

> **On 11/12/2017 at 8:13 PM, Agustin said:**
>
> I have to do something to change direct9 to dx11 or opengl? Although textures did go wrong last timme I touched something about that....

Add -force-d3d11 at the end of your KSP shortcut, after .exe.

+ Quote

Shadowmage

Sr. Spacecraft Engineer
●●●●●

Posted November 12, 2017 Report post

> **On 11/12/2017 at 8:13 PM, Agustin said:**
>
> I have to do something to change direct9 to dx11 or opengl? Although textures did go wrong last timme I touched something about that....

Members
⊕ 6,304
4,341 posts

@Electrocutor has the correct answer for activating DX11

And keep in mind that this mod, by itself, does nothing. It is simply a framework that makes the shaders available for other mods to use.

+ Quote

SS
Low Part Co
Lifters, Orbiters,
Modular Parts an
(Deve

Foozle

Rocketry Enthusiast
●●

Members
⊕ 33
72 posts

Posted November 12, 2017

Report post

This is very nice, indeed. And widely applicable which is just as important. A lot of mods could benefit from it.

Do you have any plans for a 1.3-compatible version, or would that require work beyond a simple recompile?
I ask because I'm still on 1.3, waiting for more of the mods I use to be updated to 1.31.
I already have a 1.3 version of TST and would very much like to be able to try out the Webb.

+ Quote bcink likes this

Shadowmage

Sr. Spacecraft Engineer
●●●●●

Members
⊕ 6,304
4,341 posts

Posted November 12, 2017 (edited)

Report post

> **On 11/12/2017 at 8:33 PM, Foozle said:**
>
> Do you have any plans for a 1.3-compatible version, or would that require work beyond a simple recompile?

I don't think there is anything in the plugin that would prevent it from working on KSP 1.3. You can always give it a try (AVC might complain, but it should still load and function)

(i.e. it shouldn't even need a recompile, but I haven't tested it)

Edited November 12, 2017 by Shadowmage

+ Quote

SS
Low Part Co
Lifters, Orbiters,
Modular Parts an
(Deve

vossiewulf

Junior Rocket Scientist

Members
394
660 posts

Posted November 12, 2017

Report post

> **On 11/12/2017 at 5:05 PM, harrisjosh2711 said:**
>
> My vote for modder of the year award.....

You don't know the half of it

This is NOT something Mage has been working on for a year, or even many months, he has developed this starting from a simple suggestion I made based on my own experience that he should give Substance Painter/Substance Designer a try and see if he could apply it to his mod um... **just a bit over a month ago.**

It started here on October 5th, when I posted some of my non-KSP work offering to do a little modeling as possible for SSTU, and explaining I use MAX/Flatiron/Substance Painter as my main workflow. He asked whether I thought Substance Painter was worth it, I replied here with some more examples of my work showing what could be done with an emphatic YES, anyone involved in texturing who can afford to do so should grab a Substance Painter license, they're on a reasonable subscription model now that doesn't require $200+ upfront outlay.

I gave him some very basic explanations and suggestions once he dove into Substance Painter, but it was very minimal - he's picked it up like the proverbial fish to water, and if he isn't already he'll be way ahead of me shortly. And many more good things are coming since he's diving headfirst into Substance Designer also, someone who really fully understands the tapestry of functionality that those two create can do some pretty amazing things.

So it's been zero to KSP-changing shader release in basically one month. Also keep in mind that he could have just used this for SSTU, there was no requirement to package all of this and provide the required documentation so that other modders could make use of it. **He's put in considerable work just so he could make his breakthrough available to everyone.**

So yes, modder of the year, and one of those guys who you just have to point at a problem and hand him the tools he'll need and then get the hell out of the way.

+ Quote

swjr-swis, Shadowmage, panarchist and 11 others like this

Electrocutor

Tinkerer

●●●●

Members

⊕ 600

617 posts

Posted November 12, 2017

Report post

While I am digging through the stock textures, one thing has become obvious: Squad did not make nor save the normal maps correctly. Even just opening them and re-saving them properly makes the parts look twice as good.

+ Quote

Nightside and Shadowmage like this

Foozle

Rocketry Enthusiast

●●

Posted November 12, 2017

Report post

> **On 11/12/2017 at 8:44 PM, Shadowmage said:**

Members
⊕ 33
72 posts

> I don't think there is anything in the plugin that would prevent it from working on KSP 1.3. You can always give it a try (AVC might complain, but it should still load and function)
>
> (i.e. it shouldn't even need a recompile, but I haven't tested it)

Ah, OK. Good to know.
I'm a bit leery of dll files compiled for versions higher than what I'm running-- they tend to be rather crashy.

Will let you know how it goes.

+ Quote

Shadowmage likes this

1 2 3 4 5 6 NEXT » Page 1 of 56

Reply to this topic...

GO TO TOPIC LISTING

NEXT UNREAD TOPIC

Home > Add-ons > Add-on Releases > [1.7.x] Textures Unlimited - PBR-Shader, Texture Set, and Model Loading API

Unread Threads since my last visit ✓ Mark site read

Language Theme Privacy Policy Contact Us