

Scénario Import/Export .tri dans blender (Version 2.90+)

Ce qu'il faut faire pour exporter de blender des fichiers .tri :

- Installation Blender (2.93) sur le site officiel de blender

RaydiumWikiNi BlenderVersTri

PagePrincipale :: DerniersChangements :: ParametresUtilisateur :: Vous êtes 195.221.123.14

L'export de créations réalisées avec [Blender](#) vers [Raydium](#) est possible en utilisant le script [Python](#) triEXP* livré avec les sources de Raydium.

Utilisation:

Charger ce script dans [Blender](#) ([BlenderChargerScript](#)), "triangler" l'objet dans [Blender](#) (CTRL+T, en mode édition, après avoir sélectionné tout les point avec la touche A), et [exécuter le script](#) (ALT+P dans la fenêtre qui contient le script [Python](#)). Cette opération réalisée, un sélecteur de fichier apparait dans lequel vous allez pouvoir choisir le répertoire d'export ainsi que le nom du fichier.

Vous disposerez d'un peu plus d'informations dans la console depuis laquelle vous avez lancé [Blender](#), en particulier en cas d'erreurs.

!! Attention !! vérifier que vous êtes bien en 'mode objet' avant de lancer le script.

Note: Ce script n'exporte pas le chemin d'accès aux textures, et considère donc que les textures sont toujours dans le même répertoire que le [.tri](#) exporté.

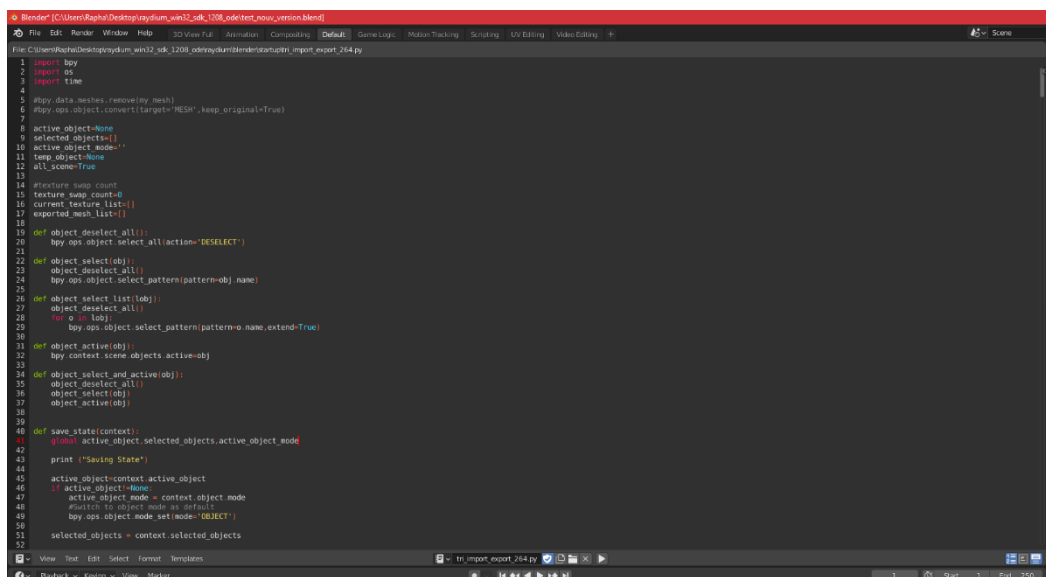
Voir aussi, pour d'autres solutions d'import/export: [ImportExportTri](#)

RaydiumWikiNi BlenderChargerScript

PagePrincipale :: DerniersChangements :: ParametresUtilisateur :: Vous êtes 195.221.123.14

- Lancer Blender
- Effacer la caméra et le plan posés par défaut (sélection avec le clic droit, "x" pour effacer)
- Faire apparaître la fenêtre "scripts" de Blender: SHIFT+F11
- Cliquer sur le "-" en bas de la fenêtre ("browse datablock") et charger le script python
- Lancer le script avec ALT+P

- J'ai donc ouvert le script dans blender après à voir triangulariser la structure.



```
1 import bpy
2 import os
3 import time
4
5 # bpy.data.meshes.remove(my_mesh)
6 # bpy.ops.object.convert(target='MESH', keep_original=True)
7
8 active_object=None
9 selected_objects=[]
10 active_object_mode=""
11 temp_object=None
12 all_scene=True
13
14 # texture temp_count
15 texture temp_count=0
16 current_texture_list=[]
17 exported_mesh_list=[]
18
19 def object_deselect_all():
20     bpy.ops.object.select_all(action='DESELECT')
21
22 def object_select(obj):
23     object_deselect_all()
24     bpy.ops.object.select_pattern(pattern=obj.name)
25
26 def object_select_list(obj):
27     object_deselect_all()
28     for o in obj:
29         bpy.ops.object.select_pattern(pattern=o.name.extend=True)
30
31 def object_active(obj):
32     bpy.context.scene.objects.active=obj
33
34 def object_select_and_active(obj):
35     object_deselect_all()
36     object_select(obj)
37     object_active(obj)
38
39
40 def save_state(context):
41     global active_object_selected_objects.active_object_mode
42     print ("Saving State")
43
44     active_object=context.active_object
45     if active_object==None:
46         active_object_mode = context.object_mode
47         switch to object mode as default
48         bpy.ops.object.mode.set(mode='OBJECT')
49     selected_objects = context.selected_objects
50
51
52
```

- A l'exécution du script cette erreur survient dans la console.

```
Traceback (most recent call last):
  File "C:\Users\Rapha\Desktop\raydium_win32_sdk_1208_ode\test_nouv_version.blend\tri_import_export_264.py", line 768, in <module>
    File "C:\Users\Rapha\Desktop\raydium_win32_sdk_1208_ode\test_nouv_version.blend\tri_import_export_264.py", line 750, in register
AttributeError: 'module' object has no attribute 'INFO_MT_file_export'
Error: Python script failed, check the message in the system console
```

-

Ligne 750 :

```
747
748 def register():
749     bpy.utils.register_class(ExportSomeData)
750     bpy.types.INFO_MT_file_export.append(menu_func_export)
751     bpy.utils.register_class(ImportSomeData)
752     bpy.types.INFO_MT_file_import.append(menu_func_import)
753
```

```
import bpy
import os
import time
```

Ce sont les modules issus du script.

Les modules issus du sdk sont soit situés dans le dossier :

1) ..\ raydium_win32_sdk_1208_ode\blender-2.64a\2.64\scripts\modules

Comme on peut le voir à la ligne 750 le module qui pose problème est bpy.

En fouillant un peu dans le sdk, on trouve plusieurs dossiers/fichiers faisant référence au bpy.

Dans le dossier : ..\ raydium_win32_sdk_1208_ode\blender-2.64a\2.64\scripts\modules

bl_118n_utils	01/02/2015 17:13	Dossier de fichiers	
bpy	01/02/2015 17:13	Dossier de fichiers	←
bpy_extras	01/02/2015 17:13	Dossier de fichiers	←
console	01/02/2015 17:13	Dossier de fichiers	
addon_utils	18/11/2012 13:14	Python File	13 Ko
animsys_refactor	18/11/2012 13:14	Python File	47 Ko
blend_render_info	18/11/2012 13:14	Python File	3 Ko
bpy_types	18/11/2012 13:14	Python File	22 Ko
bpymtl	18/11/2012 13:14	Python File	8 Ko
bpymtl_ui	18/11/2012 13:14	Python File	4 Ko
console_python	18/11/2012 13:14	Python File	11 Ko
console_shell	18/11/2012 13:14	Python File	2 Ko
graphviz_export	18/11/2012 13:14	Python File	7 Ko
keyingsets_utils	18/11/2012 13:14	Python File	7 Ko
rna_info	18/11/2012 13:14	Python File	22 Ko
rna_prop_ui	18/11/2012 13:14	Python File	5 Ko
rna_xml	18/11/2012 13:14	Python File	14 Ko
sys_info	18/11/2012 13:14	Python File	5 Ko

Copier le contenu de ce dossier dans le dossier modules de blender 2.9+ ne corrige pas le problème. Pire il empêche le bon fonctionnement de l'application.