

Instalación y uso del *script* `Tex_VTKBlender.py`

El *script* `Tex_VTKBlender.py` está diseñado para usar de manera conjunta la librería de visualización VTK (www.vtk.org) y *Blender* (www.blender3d.org), un programa de modelado, animación y renderizado en 3D.

El hecho de que desde un interprete de *Python* sea posible usar la librería VTK unido a que *Blender* cuenta con un interprete propio de *Python*, desde el que se pueden crear y modificar objetos una vez importado el modulo “*Blender*”, que contiene las funciones necesarias para ello, es lo que establece el puente de enlace entre estas dos herramientas.

1. Instalación del software

En primer lugar es necesario tener instalada la librería de VTK de manera que sus componentes puedan ser accedidos desde un interprete de *Python*. A la hora de compilar VTK es necesario tener en cuenta la versión del interprete de *Python* que se vaya a usar, ya que esta debe coincidir con la versión usada en la herramienta *Blender*. De esta manera se evitan conflictos de incompatibilidad a la hora de trabajar con VTK y *Blender* de manera conjunta.

Una vez instalado *Blender* y VTK, el interprete de *Python* necesita saber dónde están instalados el *script* `Tex_VTKBlender.py` y las librerías de VTK. Hay dos maneras de hacer que *Python* encuentre estos módulos:

➤ Usando variables de entorno

- En *Windows* creamos la siguiente variable de entorno
PYTHONPATH=\$VTK_HOME/Wrapping/Python;\$Tex_VTKBlender_HOME
- En *UNIX* según la *shell* empleada ejecutar los siguientes comandos antes del *script*
 - *Bourne*

```
PYTHONPATH=$VTK_ROOT/Wrapping/Python:${LIBRARY_OUTPUT_PATH}
PYTHONPATH=$PYTHONPATH:${Tex_VTKBlender_HOME}
export PYTHONPATH
```
 - *Csc*

```
setenv PYTHONPATH $VTK_ROOT/Wrapping/Python:${LIBRARY_OUTPUT_PATH}
setenv PYTHONPATH ${PYTHONPATH}:${Tex_VTKBlender_HOME}
```

Para evitar introducir estos comandos siempre que se inicie la *Shell*, se pueden introducir en el *script* de inicio de la *Shell*.

➤ Añadiendo las siguientes instrucciones en el *script* creado por el usuario

```
import sys
sys.path.append($VTK_ROOT/Wrapping/Python)
```

```
sys.path.append(${LIBRARY_OUTPUT_PATH})  
sys.path.append(${Tex_VTKBlender_HOME})
```

Donde:

- VTK_ROOT es el directorio donde está la instalación de VTK.
- LIBRARY_OUTPUT_PATH, es la localización de las librerías.
- BTKBlender_HOME, es el directorio en el que se encuentra el *script BTKBlender.py*

2. Funcionamiento del módulo *Tex_VTKBlender*

Tex_VTKBlender convierte objetos del tipo *PolyDataMapper* en mallas de *Blender* y mallas de *Blender* en *PolyData* de *vtk*. Para ello cuenta con dos funciones que realizan estas conversiones: *Tex_VTKBlender.PolyDataMapperToBlender* y *Tex_VTKBlender.BlenderToPolyData*.

➤ *Tex_VTKBlender.PolyDataMapperToBlender*

Esta función es la encargada de tomar como argumento un *PolyDataMapper* ya creado y generar a partir de él una nueva malla de *Blender*, en el caso de que se le pase como segundo argumento una malla de *Blender* ya creada, la función modificará esta última según el *PolyDataMapper* en lugar de crear una nueva.

A grandes rasgos, el funcionamiento interno de la función se describe en la figura 1.

➤ *Tex_VTKBlender.BlenderToPolyData*

Esta función toma como único argumento una malla de *Blender* y devuelve un objeto *PolyData* que contiene la geometría de la malla argumento.

En el *script Tex_VTK Blender* la función *BlenderToPolyData* es capaz de incluir en el *PolyData* generado la información referente a la textura mapeada en la malla original de *Blender*.

La función sólo trabaja con las texturas que han sido mapeadas haciendo uso de la herramienta “*UV/Face editor*” de *Blender*, el resto de las texturas aplicadas a la malla serán ignoradas. Esta “limitación” se debe a que desde el interprete de *Python* contenido en *Blender* solo es posible acceder a las coordenadas de las texturas mapeadas de esta manera.

Su funcionamiento es más simple que la función anterior, se resume en la figura 2.

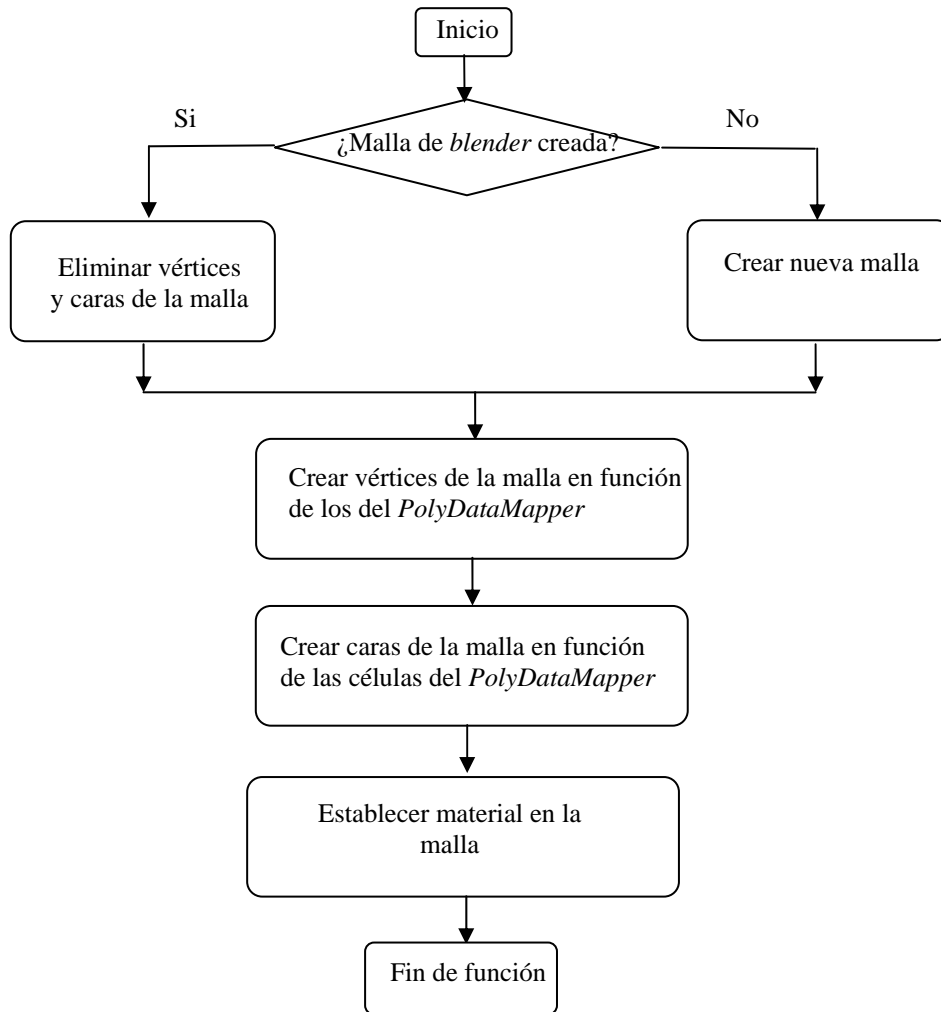


Figura 1. *PolyDataMappToBlender*

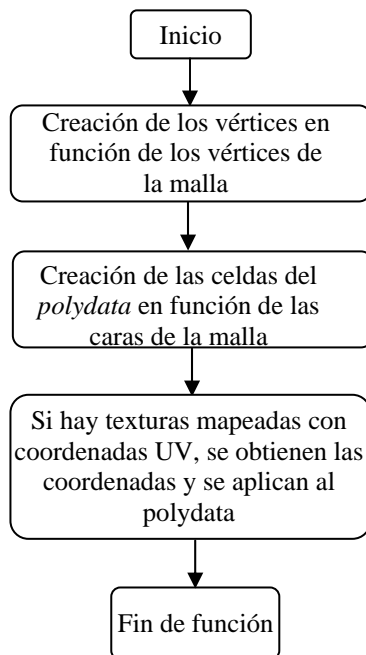


Figura 2. *BlenderToPolydata*

3. Uso del módulo *Tex_VTKBlender*

Comandos para trabajar con los módulos VTK, *Blender* y *Tex_VTKBlender*

```
include vtk
include Bender
include Tex_VTKBlender
reload(Tex_VTKBlender)
.
.
.
Tex_VTKBlender.PolyDataMapperToBlender(pmapper, me)
```

“*pmapper*” es un PolyDataMapper previamente creado
me es una malla de *Blender* previamente creada, que será modificada en función de
“*pmapper*”, es un argumento opcional, si no se incluye, se creará una nueva malla
en lugar de modificar una existente.

```
Poly=Tex_VTKBlender.BlenderToPolyData(me)
```

“*me*” es una malla de *Blender* a partir de la que se creará un *PolyData* de vtk

***NOTA IMPORTANTE:

Existe un “*bug*” en *Blender* que hace que cuando se ejecutan estas funciones en el modo “*shaded*”, se obtengan resultados impredecibles, por lo que es necesario ejecutarlas fuera del modo “*shaded*”, este modo se puede activar después de que se haya ejecutado el *script*.