

# 02 - FBX to PrimeEngine

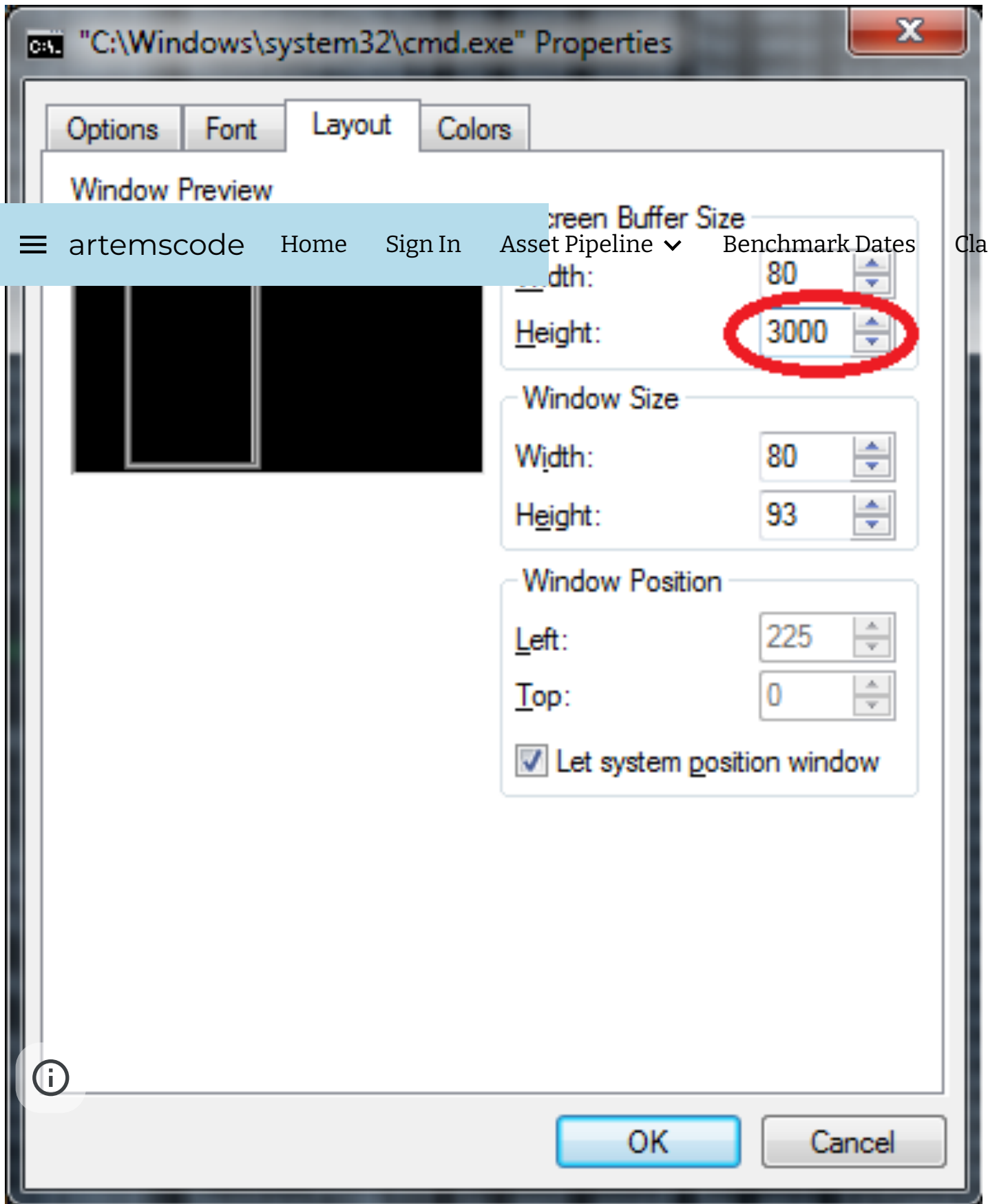
**Use the exporter stored in Tools/FBX.**

```
./fbx.sh <filename> <assetType> [-p PackageName] [-a AnimationProfile] [-flip z]
```

The parser goes through several stages, where you need to hit return. So keep hitting return until you see an error or `system.exit(0)` call.

Note, the script launches a new instance of python that you have to then manually close. The reason for it is that we are not running python within cygwin, but instead we launch native windows python that is part of PEWorkspace. Also, the reason for manually closing it is so that we could see errors that the script produces instead of automatically closing on error.

To see history of output make sure your console properties are set to store enough lines:



## Package Name

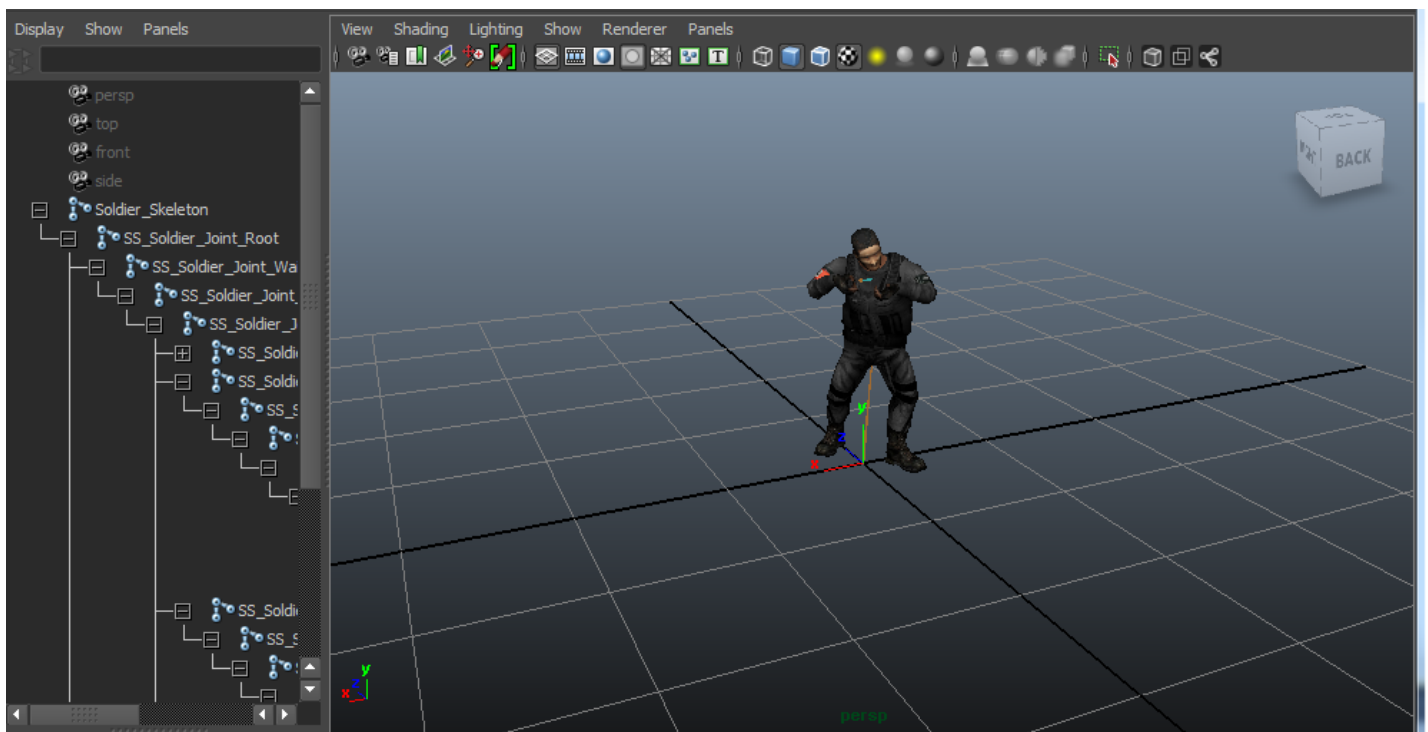
PackageName can be skipped, in which case Default is used. I suggest using custom packages to be able to easily track files.

## Flip Axis

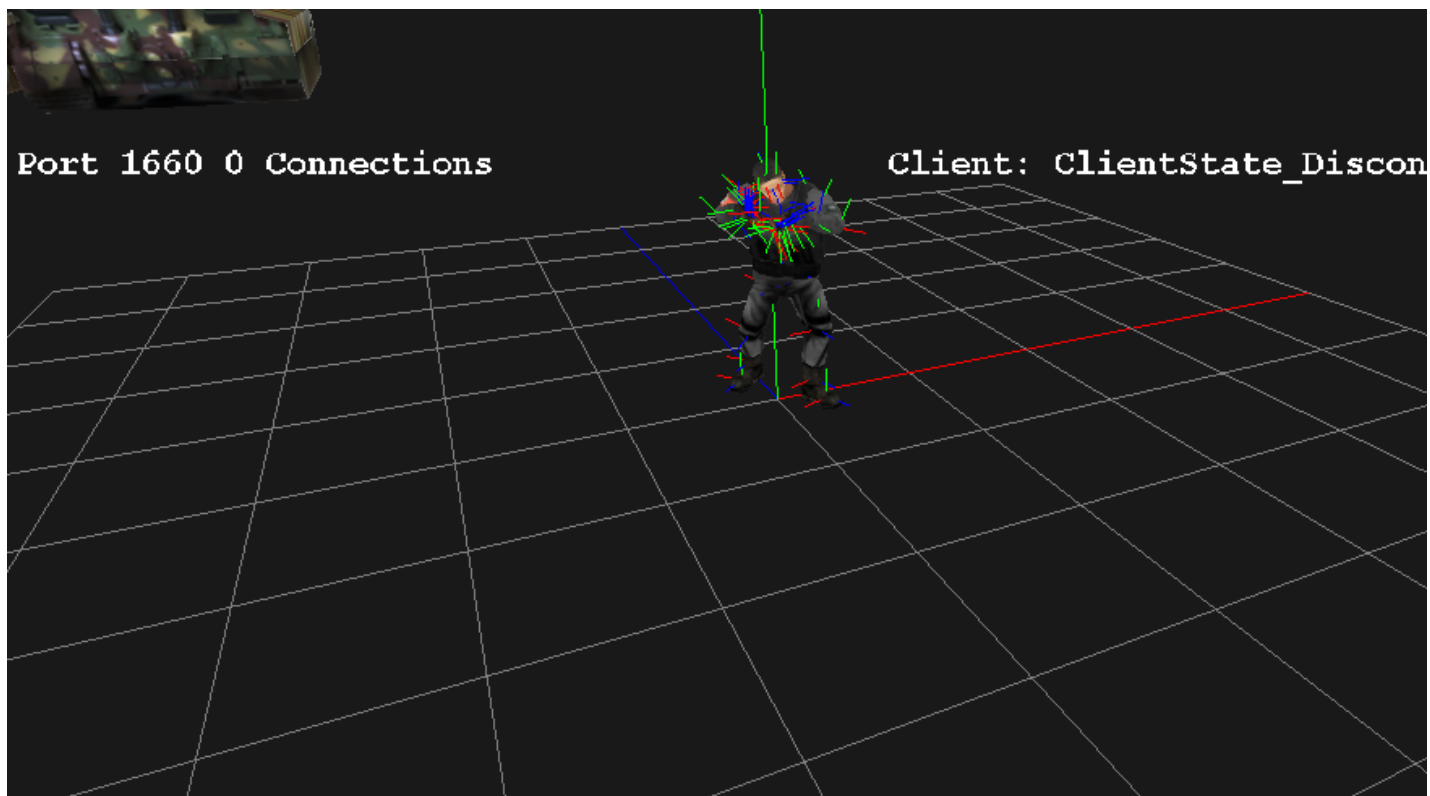
Maya is right handed, while PrimeEngine is left handed. To do that we need to flip one of the axis. -flip controls which axis to flip.

By default x axis is flipped. The other option supported is z.

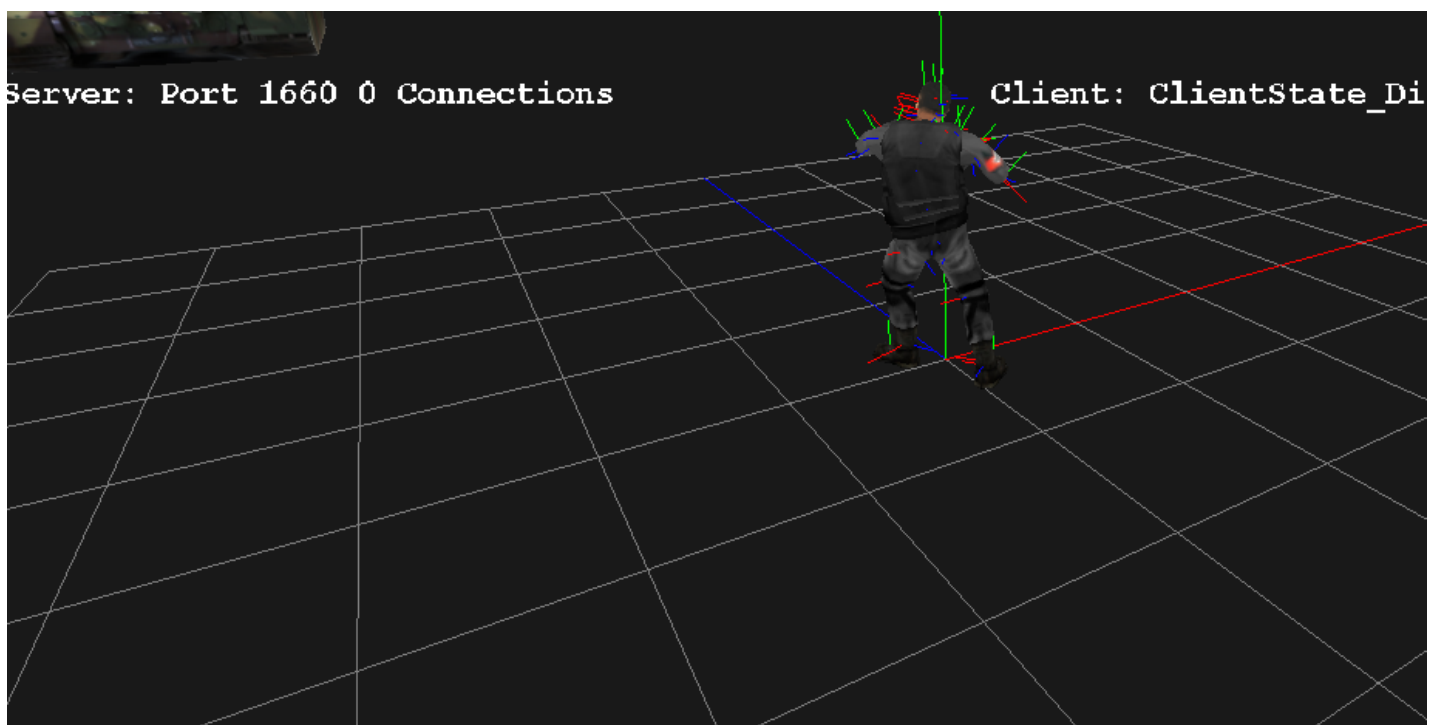
Maya Scene:



## Flip Axis X (Default)



**Flip Axis Z (-flip z)**



**Exporting Rigid Meshes**

```
$ ./fbx.sh ../AssetsIn/Maya/Soldier/soldier.fbx mesh -p TestPackage
```

Example:

```
$ ./fbx.sh ../AssetsIn/Maya/Meshes/Imrod/imrod.fbx mesh -p ImrodTest
```

After this step the mesh information is exported into the package, however:

- Textures are not exported
- Materials need to be fixed up

## Textures

The parser does not automatically copy textures referenced by source fbx file. It does try to create proper corresponding .mata files. But you have to copy corresponding dds file into Textures folder in the target package. In case of the soldier, the default package has required dds file, so we copy it (SS\_Soldier\_Final\_Texture.dds) into TestPackage/Textures

## Creating DDS Textures

Use DirectX Texture tool to create dds files. Make sure to generate mip maps. Not all texture formats work. To make sure you use correct format, open an existing dds texture (like soldier diffuse texture) in the texture tool and check its format. The usual format is X8R8G8B8.

## Material Fixup

Once textures have been copied, we need to check material files to make sure that they reference proper textures. In case of exporting imrod we AssetsOut\ImrodTest\Materials\TImrod\_blinn1.mata. We usually specify either 1 texture (standard mesh) or 3 textures (detailed mesh) or 4 textures (detailed mesh + glow)

In case of imrod we set textures like this:

4

COLOR

Imrod\_Diffuse.dds

SPECULAR

Imrod\_spec.dds

BUMP

Imrod\_norm.dds

GLOW

Imrod\_Glow.dds

You can always start out with diffuse only to make sure model works, and then add on more textures.

## Exporting Animated Skinned Meshes

Animated asset export is split into three stages

- skeleton : export skeleton
- mesh : export the mesh in t-pose (bind pose)
- animation : export the animations from file into one animation set

The reason it is split into three stages is because in engine the skeleton is the root asset of animation. Multiple meshes could be exported and skinned onto one skeleton (jacket vs pants vs hats). Similarly multiple animations can be played on one skeleton.

## Exporting Skeleton

```
$ ./fbx.sh <asset_name> skeleton -p <Package>
```

You will download a model from somewhere or build your own. Sometime you will have just the model in t-pose in one maya file, while animations will be in another (Mixamo case). If this is the case, always export the skeleton (you will need to do it once) from the t-pose file to avoid any inconsistencies across maya files that have animations in them.

As an example, a cleaned up t-pose file for mixamo vampire is:

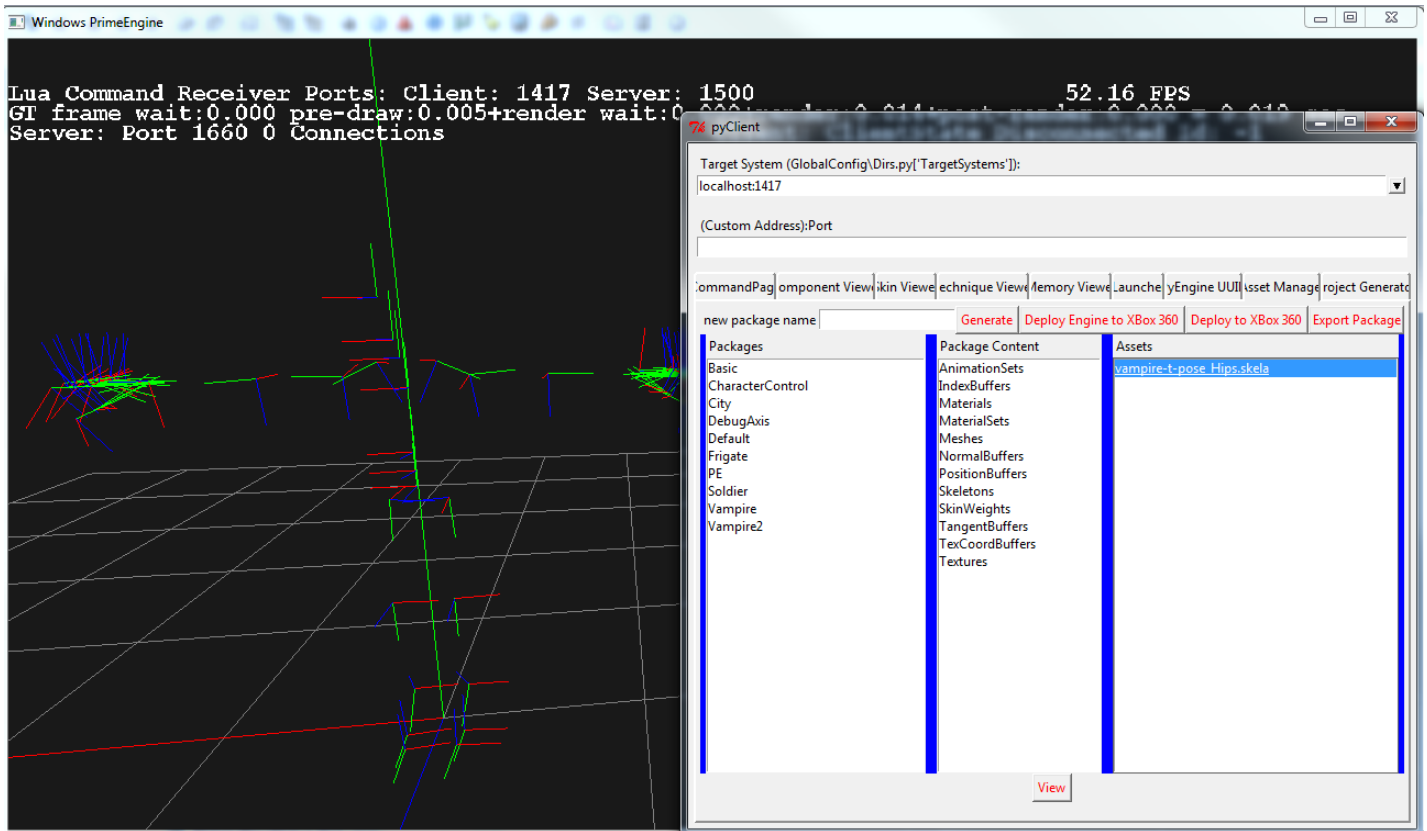
AssetsIn/Maya/Vampire/MixamoSource/vampire-t-pose-fixed-uvb.mb (loaded and cleaned up from AssetsIn/Maya/Vampire/MixamoSource/vampire-t-pose.fbx)

and exported fbx is: AssetsIn/Maya/Vampire/vampire-t-pose.fbx

To export, execute (note Vampire package already exists so we use Vampire2 for testing):

```
$ ./fbx.sh ../AssetsIn/Maya/Vampire/vampire-t-pose.fbx skeleton -p Vampire2
```

You should be able to view the skeleton through PyClient:



## Exporting Skinned Mesh

Works same as exporting normal mesh, as described above.

You don't need to mess around with the scale of the model in skinned mesh case because the exported t-pose joints carry the proper scaling. In fact what happens is that if the model is still big, then joints are very small (size of 1cm or 0.01m, when debug-rendered in engine they are scaled 100x)

As an example, a cleaned up t-pose file for mixamo vampire is:

AssetsIn/Maya/Vampire/MixamoSource/vampire-t-pose-fixed-uv.s.mb (loaded and cleaned up from AssetsIn/Maya/Vampire/MixamoSource/vampire-t-pose.fbx)

and exported fbx is: AssetsIn/Maya/Vampire/vampire-t-pose.fbx

To export mesh we do (note Vampire package already exists so we use Vampire2 for testing):

```
$ ./fbx.sh ../AssetsIn/Maya/Vampire/vampire-t-pose.fbx mesh -p Vampire2
```

(don't forget to fixup materials and copy textures)

Result material for vampires look like this:

```
AssetsOut/Vampire2/Materials/Vampire_Vampire_MAT.mata
```

```
MATERIAL_V1
```

```
0.000000 0.000000 0.000000 1.000000
```

```
0.000000 0.000000 0.000000
```

```
0.500000 0.500000 0.500000
```

```
0.500000 0.500000 0.500000
```

```
4
```

```
COLOR
```

```
Vampire_diffuse.dds
```

```
BUMP
```

```
Vampire_normal.dds
```

```
SPECULAR
```

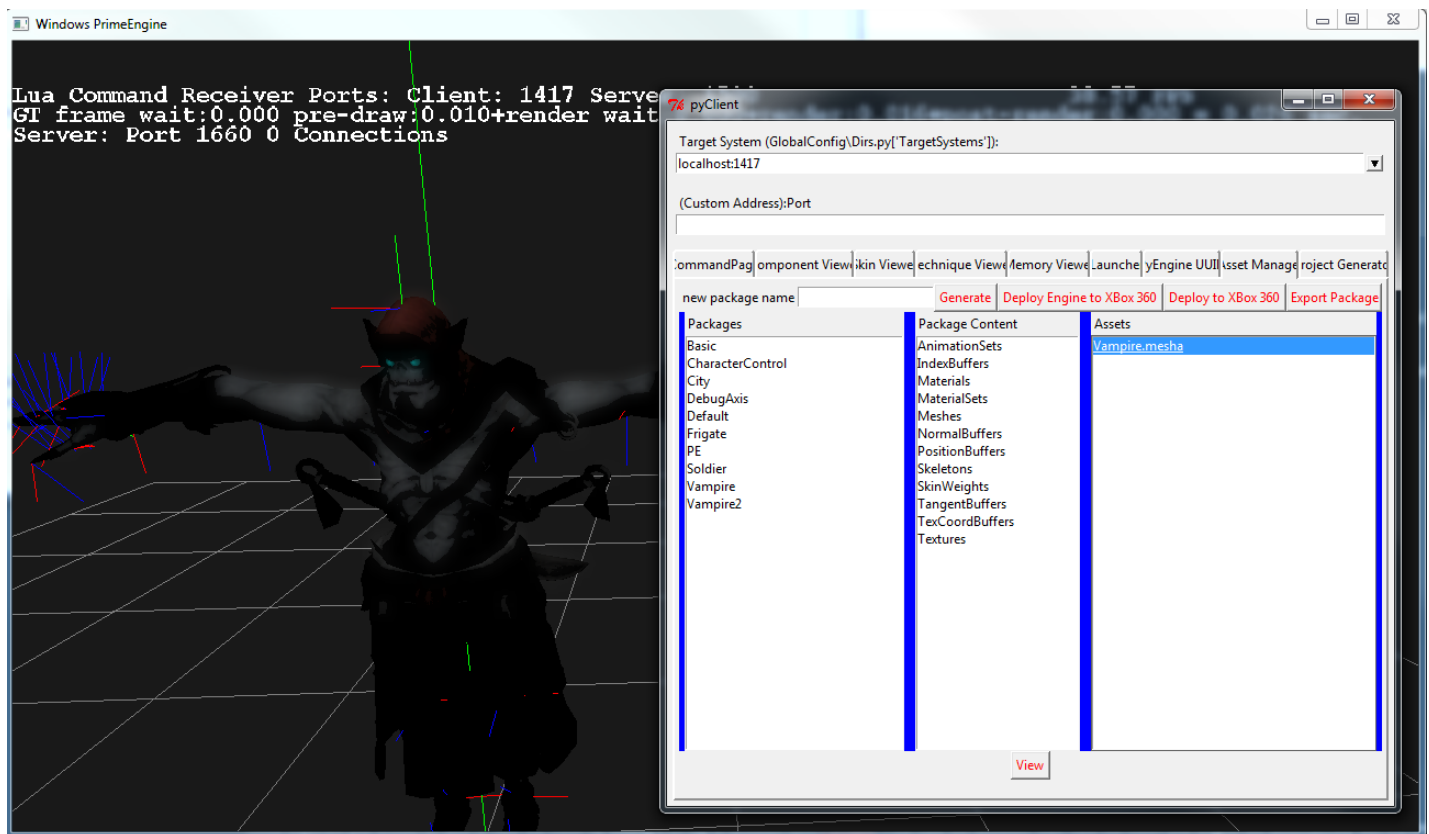
```
Vampire_specular.dds
```

```
GLOW
```

```
Vampire_emission.dds
```

Once exported you should be able to view the mesh by adding it on top of skeleton form PyClient. First view the skeleton, then view the mesh (the engine will automatically add the mesh to skeleton):





## Exporting Animation

```
$ ./fbx.sh <asset_name> animation -p <Package> [-a <AnimationProfile>]
```

You might have one maya file (then exported to Fbx) with all animations or one maya file per animation. When PE Fbx Parser processes the fbx it will store the animation data into an animation set file. In engine you can have multiple animation sets loaded on a skeleton.

### Animation Profile

Maya Fbx exporter exports animation into a single animation. I use a .profile file to tell our fbx parser how to separate the single animation into a set of animations.

Animation profile file is a file with a list of animations. Here's an example for downloaded action adventure animation pack fbx from mixamo:

```
d={
'sets':[
["a00 idle look around" "2" "122"]
```

[ "a00\_idle\_look\_around", 2, 122 ],

["a01\_falling\_idle","123","144"],

["a02\_idle","145","445"],

["a03\_stomp\_r\_foot","446","544"],

["a04\_run\_root\_move","545","566"],

["a05\_stand\_to\_cover\_l","567","601"],

["a06\_walk","602","633"],

["a07\_stand\_to\_crouch\_cover\_l","634","672"],

["a08\_run\_to\_stop","673","700"],

["a09\_stand\_turn\_left","701","732"],

["a10\_stand\_turn\_right","733","764"],

["a11\_stand\_jump\_up","765","772"],

["a12\_jump\_to\_roll\_fw","773","827"],

["a13\_jump\_to\_stand\_up","828","888"],

["a14\_crouch\_cover\_l\_to\_stand","889","934"],

["a15\_crouch\_cover\_l\_idle","935","1012"],

["a16\_crouch\_cover\_l\_walk\_l","1013","1051"],

["a17\_crouch\_cover\_r\_walk\_r","1052","1085"],

["a18\_cover\_l\_walk\_l","1086","1120"],

["a19\_cover\_r\_walk\_r","1121","1155"],

["a20\_cover\_l\_idle","1156","1225"],

["a21\_cover\_l\_to\_stand","1226","1267"],



```
'children': [child_joint0, child_joint1, ...]
```

```
'anim': {
```

```
  "anim_name0": [
```

```
    #anim keys
```

```
    {
```

```
      'frame': start_frame,
```

```
      'tx': float_tx,
```

```
      'ty': float_ty,
```

```
      'tz': float_tz
```

```
      'rx':
```

```
      'ry':
```

```
      'rz':
```

```
      'sx':
```

```
      'sy':
```

```
      'sz':
```

```
    },
```

```
    ...
```

```
  ]
```

```
}
```

```
'frameToAnimKeys': {}
```

```
}
```

## Post Rotation

FBX format has pre-rotation and post rotation.

Pre-rotation corresponds to joint orient,

Post-rotation corresponds to rotate axis, but inverted

<http://forums.autodesk.com/t5/FBX-SDK/Maya-quot-Rotate-Axis-quot-vs-FBX-quot-PostRotation-quot/td-p/4168814>

