

Saints Row IV SDK Release A: Weapon Mod Tools



Supported

Replacing existing weapons in Saints Row IV with a custom model and new textures.

NOTE: The replacing of existing assets are temporary. Volition is working on a modding patch for Saints Row IV that will allow the loading of entirely new assets without replacement! No ETA as of yet, but get hyped.

Currently Unsupported

Adding entirely new weapons. (Only replacing existing.)

Adding new effects to weapons. (Particles, etc.)

Adding new animations to weapons.

Creating Wacky Waving Weapons! (Yep, you know the ones.)

Beyond these tools, you'll need the following external tools:

Python 2.6.6

Wx Python

FBX Python SDK 2014 for Python 2.6

Autodesk 3DSMax OR Autodesk Maya 2014

Minimaul's Saints Row IV Tools

SECTION 1: DOWNLOAD EXTERNAL TOOLS

I haven't tested it personally, but if you are on a 32 bit operating system, download the 32 bit versions of each program if applicable.

Python 2.6.6

Download the 64 bit version named Windows X86-64 MSI Installer

Wx Python

Grab the one named "wxPython3.0-win64-py26"

FBX 2014.1 SDK

Install all 3 programs, with Python 2.6 being installed first.








Finally you need to copy some files from the FBX 2014.1 SDK directory to your Python 2.6 directory. This [link](#) from Autodesk explains it pretty well.

Minimaul's Saints Row IV Tools*

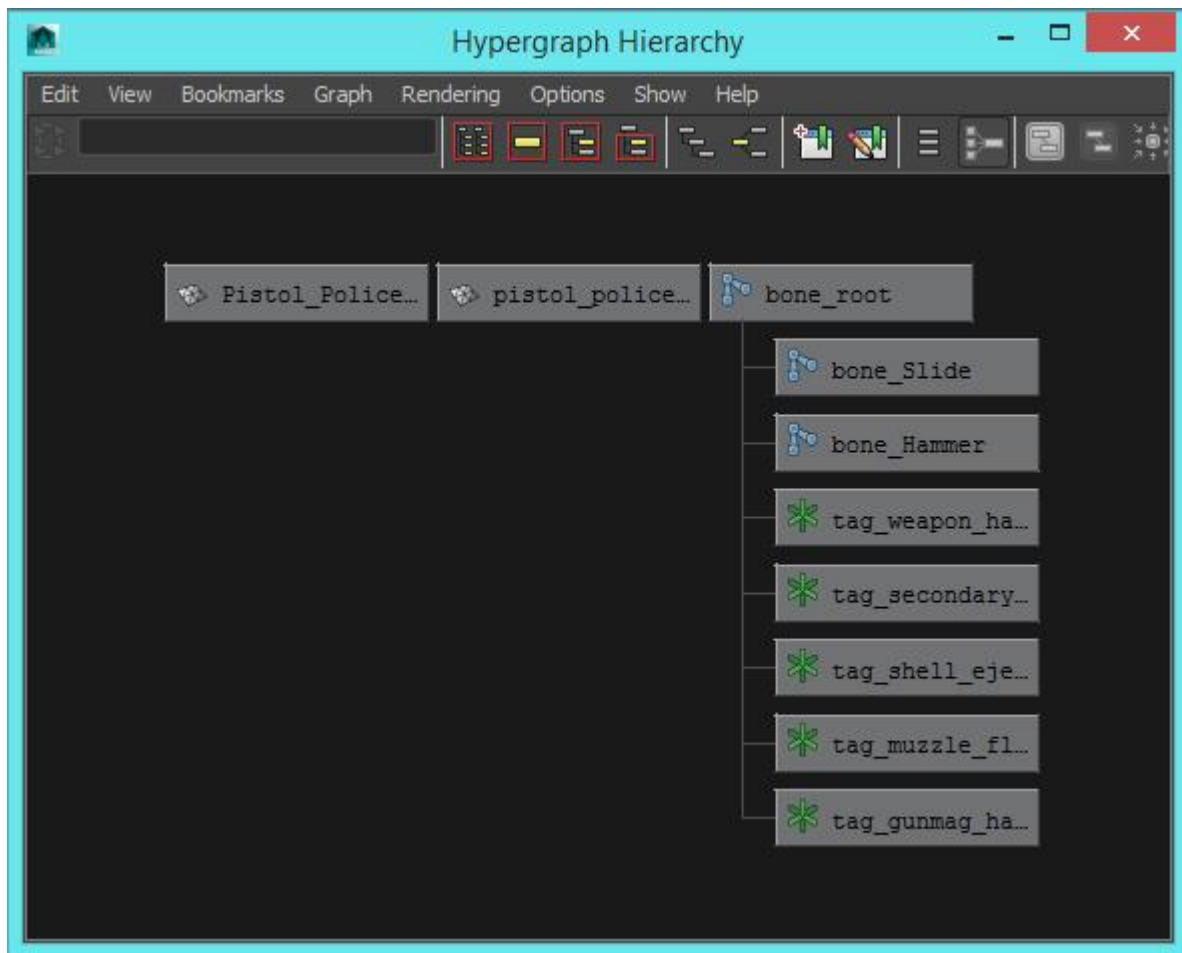
**This download is technically optional. This is for those of you that would like to replace weapons not included in the templates.*

SECTION 2: TEMPLATES

Extract your 'templates' folder. These folders have examples of each weapon style in the game. These FBX files contain everything you need to make a weapon replacement for Saints Row IV.

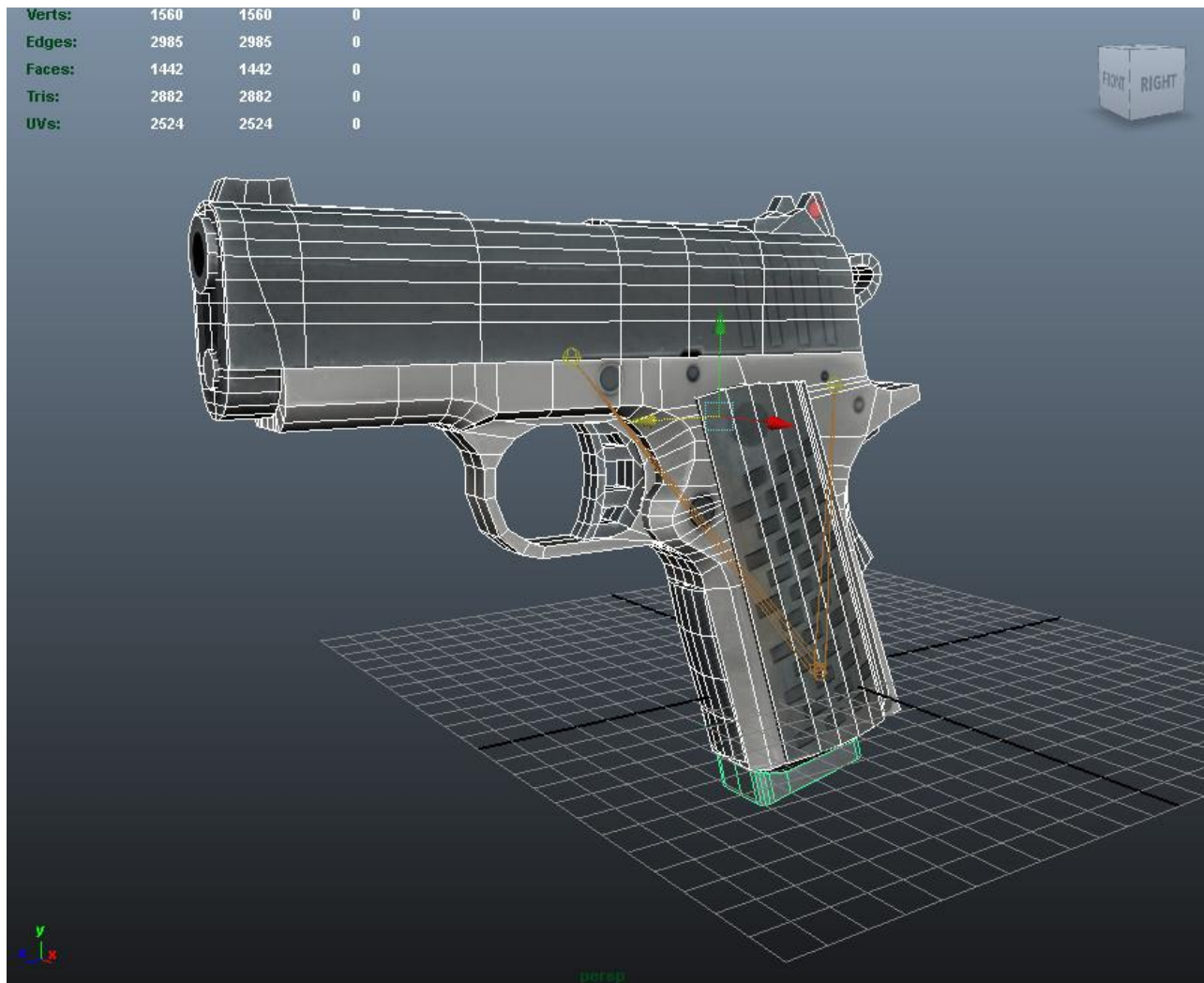
| SaintsRowIV > sr4_modding_private.git > tools > FBX > SR4_Weapon_Templates > | | |
|---|-------------------|-------------|
| Name | Date modified | Type |
|  Heavy | 5/20/2014 5:43 PM | File folder |
|  Melee | 5/20/2014 5:43 PM | File folder |
|  Pistol | 5/20/2014 5:43 PM | File folder |
|  PumpShotGun | 5/20/2014 5:43 PM | File folder |
|  Rifle_SMG | 5/20/2014 5:43 PM | File folder |
|  RPG | 5/20/2014 5:43 PM | File folder |
|  Underslung | 5/20/2014 5:43 PM | File folder |

Template Contents



The templates include several things. First the mesh for the main gun, clip (for some weapons) and then bones and locators. I've laid out the specifics of each below.

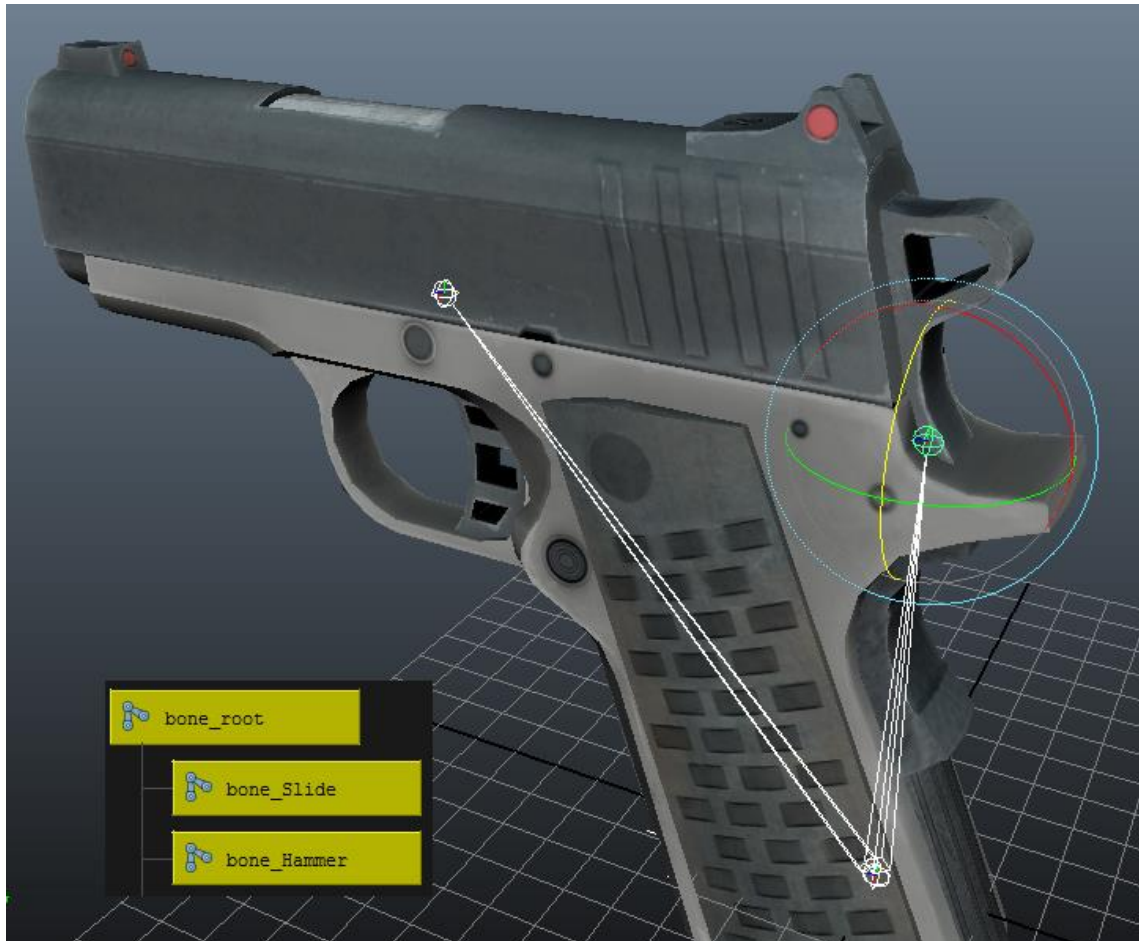
Meshes



In the case of the Pistol template, 'Pistol_Police_A' is your weapon. Currently it is skinned to the bones, and has a shader applied to it. The name of the mesh is important to note, this is what you will need to name your new mesh. The name of the mesh is also used by the converter to name the files.

Some of the weapons have an additional mesh, which is either a clip or a projectile. In this case we have the "pistol_police_a_bone-clip" mesh. These have the same shader applied to them as the main weapon and are not skinned to a bone.

Bones

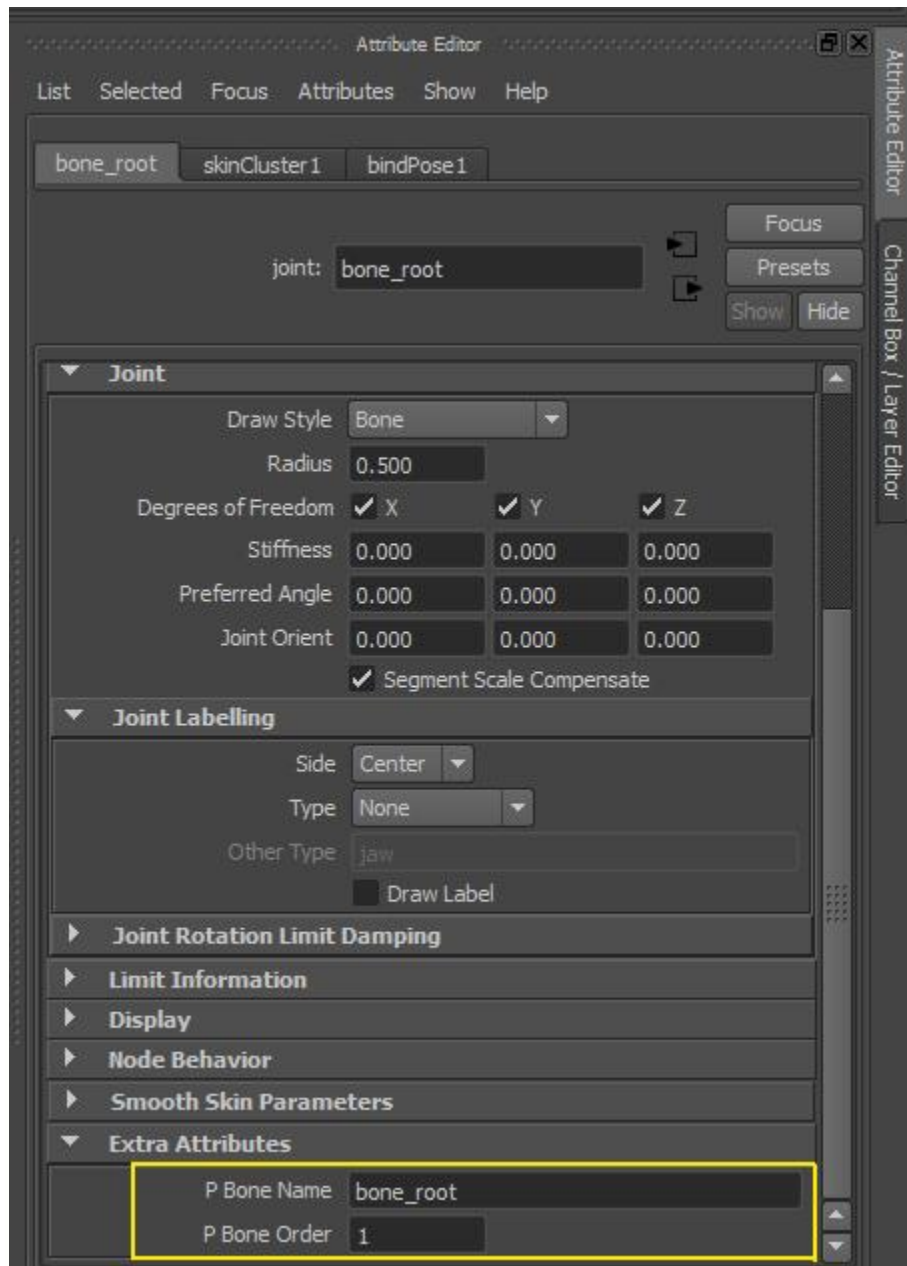


Bones are used to specify what areas of your weapon animate. The names are usually pretty self-explanatory. When you rig the skeleton to a new mesh, it will animate in the same way as the template weapon. For example, the slide bone on the Pistol template moves its skinned vertices back in the Z axis when fired.

Another example is the Mine Launcher's BlueCanister bone, any vertices skinned to it will rotate in the X axis. There is currently no way to change how these bones behave, and weapons can only use bones they originally had.

Clips / Projectiles and melee weapons are not skinned to a bone. These will be converted into an **.smesh** file in the converter. (Static Mesh)

Meshes that are skinned to a bone will be converted into a **.ccmesh** file in the converter. (Character Mesh)



Bones also have a couple extra attributes. These are necessary for the bones to be properly read by the game. You likely won't have to bother with these settings since they are already set correctly in the template files.

Attachment Tags



Attachment tags are null objects that define where attachment points and various effects like particles are located. Tags are defined by name, with the prefix of 'tag_'.

The templates contain all the tags for that particular weapon, so all you will need to do is move their position to the appropriate place on your new mesh.

MAYA NOTE: The latest Maya FBX Plugin exports these nodes as empty groups that have no graphical indicator. Select them from the Hypergraph and then select the translation tool in order to see them. (The key shortcut is W.)

If you want to replace some of the other weapons not included in the templates, here are some mandatory tags that we can use. Listed below.

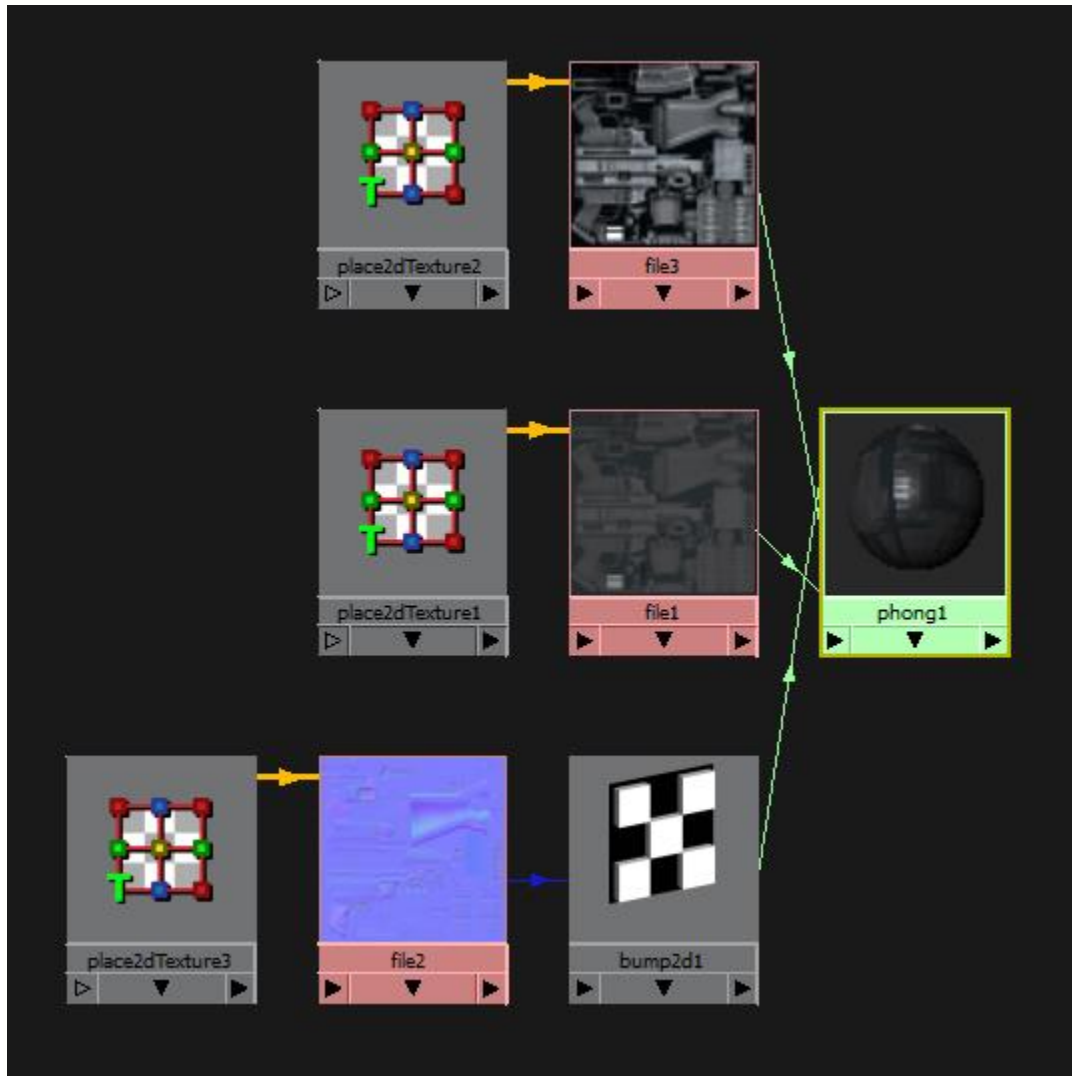
weapon_handle - Where it attaches to the hand of a character

secondary_handle - Where the offhand will IK attach to the weapon (IK = inverse kinematics)

muzzle_flash - Where the muzzle flash FX will play from

NOTE: Currently we don't know the other tags and bones that the other non-template weapons have. But it is in the plans to provide the rest of them in the future. Basically, you can replace all the weapons currently but until you know the correct bone setup and tags, you can't replicate all features. For example, you can replace the Tentacle Bat, but it won't have the physics because we don't know the names of the bones and locators. The template files have been created from source files from the artists at Volition.

Materials



Above: Pistol's Shader opened in Maya's Hypershade.

- Textures are assigned using Maya / 3DSmax shader network.
- Textures need to be saved as .TGA format.
- Converter Supports Diffuse, Normal, Glow and Specular Maps.
- Converter does not currently support multiple materials.
- Texture names need to follow a set naming convention for the FBX converter to be able to find them. The naming convention is laid out below.

texturename_SM_D.tga

texturename_LG_X.tga

X = (D, N, or S. These stand for Diffuse, Normal or Specular.)

LG = Large Texture used for up close.

SM = Small texture used on LOD models.

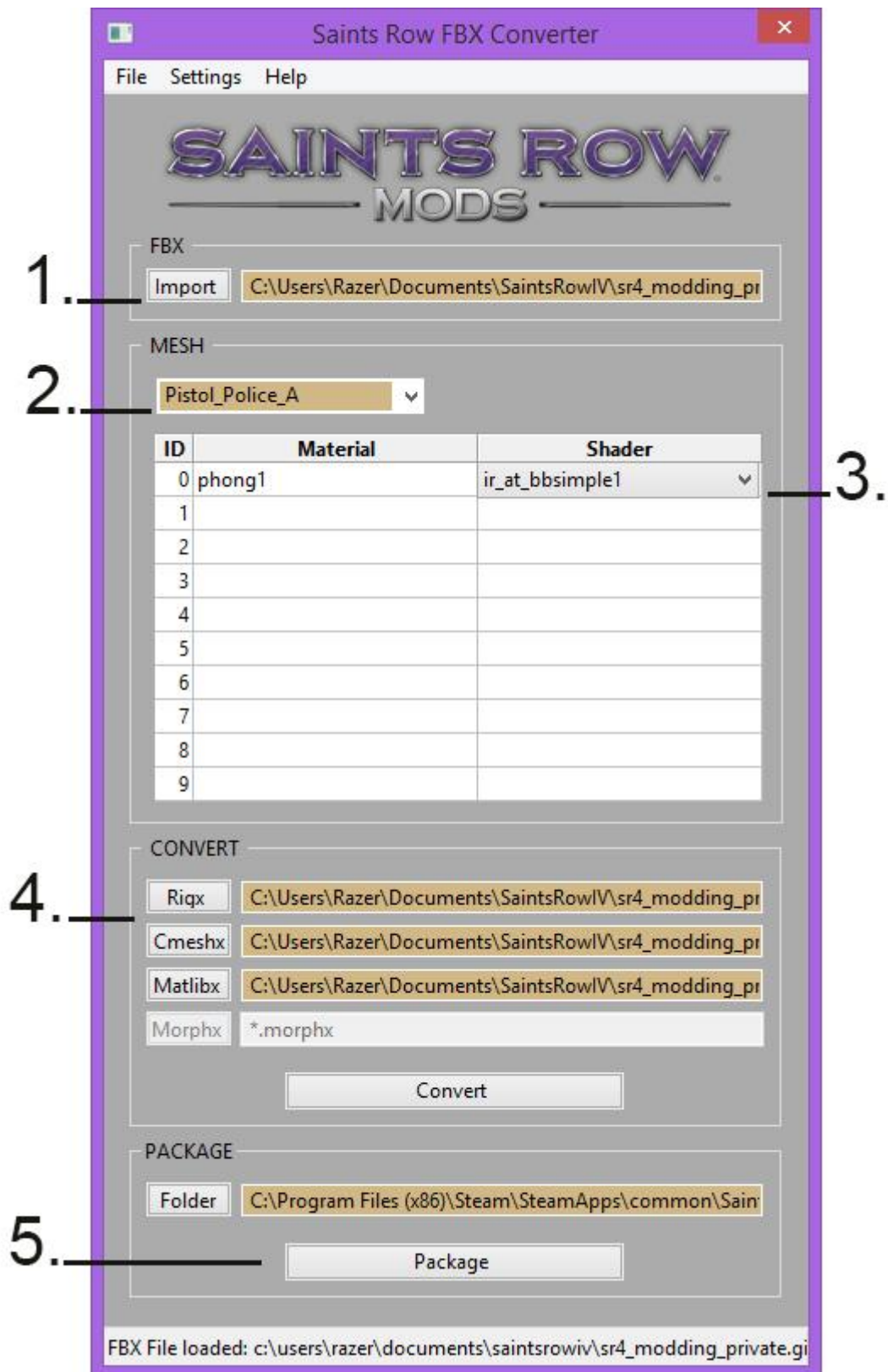
- Your diffuse texture for *texturename_SM_D.tga* should be 512x512 pixels in dimensions.
- Your LG normal and diffuse should be 1024x1024 and your LG specular map should be 512x512. Keep in mind that you don't necessarily need anything but a diffuse.
- Textures need to be in the same folder as the FBX.

STOP: At this point we want to create our FBX.

Open the Maya Text Tutorial.

And once you are done, come back to this part to continue.

SECTION 3. CONVERTER



Now launch the FBX converter. I'll lay out what it does.

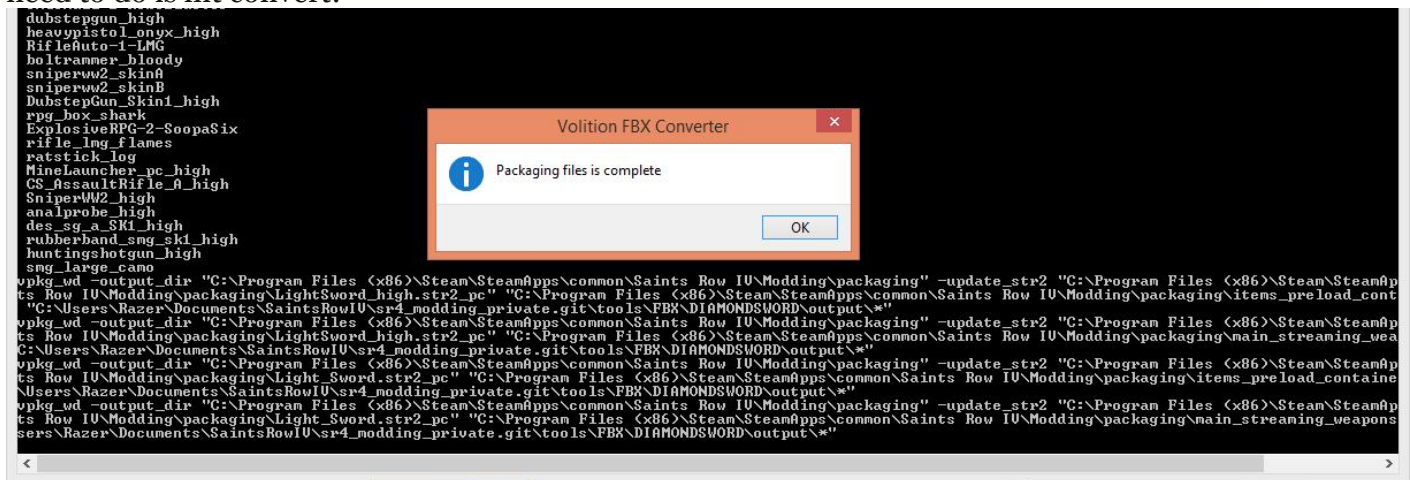
1. This one's pretty obvious, navigate to your FBX and select it.
2. This dropdown menu lets you select between each mesh in your FBX, in this case we can select the main weapon mesh and the clip mesh of the Police Pistol.
3. Select '**ir_at_bsimple1**' as your shader. The rest of the shaders aren't supported / documented.
4. These toggles enable export / converting of various file types. You want to enable all of these for your main mesh.

If you import something like a melee weapon or select a clip from the dropdown you will notice that it disables Rigx. This is because these are crunched as static meshes instead of character meshes. (No bones remember!)

5. I will talk more about packaging in the next section. This is our last step when making a weapon, and this button will take our crunched files from step 4 and package them up for the game. Again, more details in the next section.

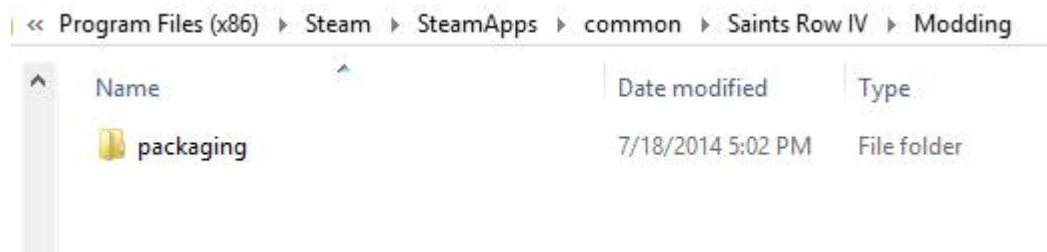
REMEMBER: If you are making a gun with a clip / projectile, it most likely had a dash in the name. Maya doesn't preserve dashes in the name, so right now is the moment to change the underline to a dash.

So, you've imported your FBX, and you've enabled the buttons detailed in 4. Now all you need to do is hit convert!



SECTION 5. PACKAGING FILES

Directory Setup



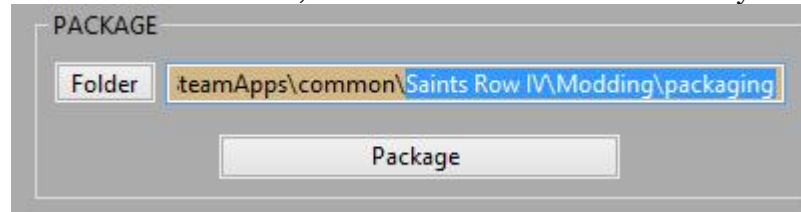
Create a folder named **packaging**, inside a folder named **Modding**.

<..\Steam\SteamApps\common\Saints Row IV\modding\packaging>

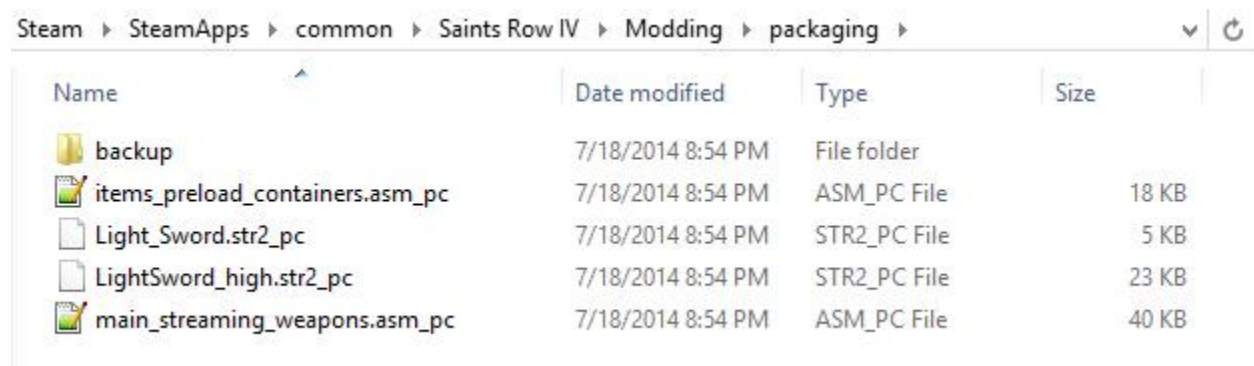
Packaging is where you put the STR2 / ASM files of your chosen template. All you have to do is copy and paste the files from the '**Packfiles**' folders included with each template as seen below.

| | | | |
|-------------------------|--------------------|-------------|----------|
| output | 7/9/2014 1:02 AM | File folder | |
| packfiles | 7/18/2014 5:50 PM | File folder | |
| huntingshotgun.FBX | 7/14/2014 2:02 PM | FBX File | 87 KB |
| HuntingShotgun_lg_D.tga | 12/21/2012 9:04 AM | TGA File | 3,073 KB |
| HuntingShotgun_lg_N.tga | 12/19/2012 1:29 PM | TGA File | 769 KB |
| HuntingShotgun_lg_S.tga | 12/21/2012 9:04 AM | TGA File | 3,073 KB |
| HuntingShotgun_sm_D.tga | 12/19/2012 3:47 PM | TGA File | 193 KB |

Back in the converter, hit the folder button and select your **packaging** folder.



All you need to do now is hit Package!



If all went correctly, then you will have your modified files in your **packaging** folder! Take them and shove them in Saint Row IV's root folder (C:\Program Files(x86)\Steam\SteamApps\common\Saints Row IV) and launch the game!



If everything went right, your new weapon will replace the default model. Celebrate!