Maya to Source Model Importing – 5 Minute Workflow

By Rob Howland
Edited by Matt Forehand

I have devised a method that utilizes many third party applications and plug-ins that will enable any serious development team to import Maya created models into the source engine in roughly 5 minutes time. This is the definitive tutorial for Maya to Source importing, and by the end of these easy to follow step-by-step instructions, you will be an important asset to any team utilizing Source. So without further ado…

Programs Needed:

- The Maya Exporter
  - Prall's Maya to HL2 Export Script (absolutely essential – thank this guy)
    - http://mitglied.lycos.de/prallvamp/hl2export2.5.zip
- The .QC File Compiler
  - GUIStudioMDL (best interface – extract to your BIN folder – see below)
    - http://www.wunderboy.org/download.php?file=guistudiomdl_2.2.zip&s=2
- The Texture Converter
  - 360g (has normal map issues – extract to your BIN folder – see below)
    - http://www.map-lounge.de/zips/360e.zip
- The Normal Map Solution
  - Photoshop VTF plug-in (can be used in place of 360g but slows workflow)
    - http://nemesis.thewavelength.net/files/files/psvtfplugin1011.zip
  - NVIDIA Photoshop Normal Map Plug-in
    - http://developer.download.nvidia.com/tools/texturetools/Photoshop_Plugins_8.23.0307.1915.exe
- Other Useful Source Modding Tools
  - GCFScape - Nem's Wonderful .GCF Archive Reader and Extractor
    - http://nemesis.thewavelength.net/files/files/gcfscape170.zip
  - Crafty – Nem's Amazing Map & Model Viewer – Exports Source OBJs!!!
    - http://nemesis.thewavelength.net/files/files/crafty100alpha15.zip
  - VTFLib/VTFEdit – Another of Nem's Tools – Great VTF/VMT Tool!
    - http://nemesis.thewavelength.net/files/files/vtfedit125-20.zip
  - MDLDecompiler – CannonFodder's .MDL File Decompiler – Duh!
    - http://www.chaosincarnate.net/cannonfodder/download.php?id=mdldecompiler.05.rar

Step 1 – Setting up Your Workflow and File System

Once you have installed all of these applications, you will have all the tools you need to get your models into the Source Engine. Extract both the GUIStudioMDL and the 360g programs into your BIN folder, which is located:

"C:\Program Files\Steam\steamapps\YOURUSERNAME\sourcesdk\bin."

Place a shortcut to the BIN folder somewhere very accessible as you will be navigating here quite often. Create a new favorites folder and place it on your explorer window.

The Maya exporter developed by Prall is temperamental. It needs everything to be in order for it to export your model correctly. You will need to make a commitment to one of the Valve games on your system. We are going to use Half-Life 2 as the basis for this tutorial.

Navigate to your "**sourcesdk_content**" folder located in the "**YOUR-USER-NAME**" folder and place a shortcut to that "**sourcesdk_content**" folder in the very same place as your BIN shortcut. Because we are using Half Life 2 in this tutorial select the hl2 folder. If not, just choose the game you want. Do not worry if you plan to import models into multiple games, this step is only important because of how picky Prall's exporter is.

You will still be able to import your models into any source powered game; the only catch is that all your textures, all your Maya scene files, and all your exported stuff WILL END UP IN THIS FOLDER. You still get to choose which game to place it in later on down the line so fear not.

Once inside the game folder, there will be three more folders, "**mapsrc**", "**modelsrc**", and "**materialsrc**". The first isn't used in the process but the other two are crucial. The "modelsrc" folder is the one you will be using to save your scene files in and export your models.

*Editor's Note: The "**materialsrc**" is the one you will NEED to put all your TGA texture files into in order for the exporter to retain the texture attributes after exporting. This is the crucial step that if you do not follow will give your model the pink and black texture issue. REMEMBER THIS!*

Now navigate back to that "**YOUR-USER-NAME**" folder and find the corresponding game folder to the one you have decided upon. For half-life 2, navigate to "**half-life 2**" and place a shortcut to the "**hl2**" folder in the same spot as your **BIN** and content folders. This is where all your final compiled models and converted textures will be going after you are successful in exporting them. So now that you have those three shortcuts handy, you won't be wasting time flipping through your file system trying to find the folder you need. These will help you greatly.

Optional Step (should be done your first time through):

*The final part is to open Maya and set up a project. To do so, go to File => Project => New and name your project "**hl2**" (or **cstrike**, or **dod**, or whatever you decided upon). Now you need to navigate to that "**sourcesdk_content**" folder located in **"C:\Program Files\Steam\steamapps\YOUR-USER-NAME\sourcesdk_content"** in case you forgot. Now rather than selecting the "hl2" folder, you must select the "**sourcesdk_content**" folder so that it will place the project folder that you named **"hl2"** on top of the existing **"hl2"** folder. Instead of clicking "**use defaults**", there are only two fields you need to fill in. For Scenes, type "**modelsrc**" and for Source Images, type "**materialsrc**".*

*Since these folders already exist, you will not notice any changes in your file system, but Maya will now recognize that there is a project here. If for some reason you need to set the project again in the future, simply go to **"File => Project => Set"** and navigate to the **"hl2"** folder.*

I am assuming that you actually have a custom model ready to be imported into Source. I am also assuming that you know the basics of Maya and how to actually create a finished piece without non-manifold geometry. If you are skeptical about your modeling abilities, I highly recommend using Maya's optimization tools.

There is no need to triangulate the model as the exporter handles it.

*One little step that will save you time is to set Maya's primary axis to "**Z-up**". Go to **"Window => Settings/Preferences => Preferences"** and click on **"Settings"**. Where it says **"World Coordinate System"** set it to "**Z**".*

*Set the **"Working Units"** labeled **"linear"** to inches. A character's height in Source is roughly 80 inches.*

You will need to do a number of steps before you are ready to export. These can be real time eaters, so I have added them to that very handy HL2Export toolbar in order to make it easier to get models out quickly. These little steps are "Freeze Transformations", "Center Pivot", and "Delete History".

*Adding these features to your toolbar is not something every Maya user is familiar with so I will explain how to do so. With your HL2 shelf open, go to "**Modify**" and then you are going to hold down CTRL+SHIFT and click on "**Freeze Transformations**". Doing so will add "**FT**" or "**Freeze Transformations**" to your toolbar. Do this with the remaining buttons.*

Step 2 – Exporting Your Model

Once the model is open in Maya and your Targa texture is in place, it is now time to explore the HL2Export toolbar located on the Maya shelf. Under the HL2Export tab there are four icons that say **"Expor"**, **"ADD", "NEW",** and "**Layer**".

"**ADD**" creates a Phong E shader.

"**NEW**" creates Shading Group.

"**Layer**" Allows you to create the physics and the reference layers.

"**Expor**" opens up the export window.

It is important to place the center of mass on the origin so you can view your model in the Source model viewer. *The best way to find the center of mass is to use your Center Pivot button, which will place your pivot in the center of mass*. Once you have centered your model on the origin, freeze transformations to reset the values to zero without moving the model.

Once you have frozen the transformations, duplicate your model. Do not move this model as it will be important later in the process.

A brief note about animations:
Do not freeze your transformations. Do not delete your history after you have keyed your animations or they will be destroyed. Just be very careful while you're modeling and make sure you clean your model before you finalize it for animations.

Once the model is open in Maya and your Targa texture is in place you are ready to begin the export process. First, open the scene file in Maya and immediately go to SAVE AS and save another copy of the model in your **"modelsrc**" folder. It is critical that the scene is saved here for the exporter to work correctly. Now place the model's texture file in your "**materialsrc**" folder. The format **MUST** be **Targa (*.TGA).**

Use the "**ADD**" button to create a new Phong E and apply it to just one of your models. Apply the color map saved in your "**materialsrc**" folder.

Important Note: Normally (pun intended) the model will have multiple texture maps, so you must create multiple Phong E shaders for each different texture.

Editors' Note: *This may be what the Shader Group is for but we don't really know. Any information that could be provided as to the actual function of this would be appreciated and accredited.*

Use the "**Layer**" button to bring up the Create Layers Window and click on the "**ref/phy Layer**" button. This button creates two empty layers inside Maya labeled "**Physics**" and "**Reference**". Select your textured model and add it to the "**Reference**". Now add the duplicate that you made from early and add it to the Physics layer. This layer should not have any texture applied to it beyond the Maya defaults.

Important Note: Not all models are this simple. The Source engine works by displaying the reference model and at the same time, using an invisible model for Physics. This allows for the physics model to take any shape you find necessary.

The compiler will try to cut down as many polygons as possible of the physics model in order to increase performance. You can turn off this feature, and it will use all the polygons to make a "full collision model" but this is not recommended for performance reasons.

For a complex model, any simple model of the basic geometric shapes could be used in place of the physics model. This feature is critical if your reference model is high-poly or contains convex areas. If you choose this method, delete the copy of the model on the **"Physics"** layer and model the new simplified geometry, and add it to the physics layer. Doing so can give you more control making physics objects move and react the way you intended them to.

Hide your reference layer, and then select your physics layer and **SOFTEN EDGES**.

Important Note: "Window => Relationship Editors => Display Layers." Now on the left you will see your layers (reference and physics) and on the right you will see Model Nodes. Expand the model nodes (click the "+" symbol) so that it displays the actual mesh node (the blue square) with the white slanted rectangle above it. All you have to do is select the reference layer on the left, which will show you its contents on the right. The blue square is not highlighted; therefore you must select it by clicking it, adding it to the layer. Do the same thing to the physics layer.

Restore your reference layer, select both objects, and delete your history.

You are now ready to click the "**Expor**" button. The Export window will pop up. DESELECT all the options except the first two and the two marked "**export 1 frame idle smd**" and "**generate QC**".

Click the tab at the top labeled "**path settings**" and click the "**default**" button to fill in the fields.

Editors' Note: If the fields stay blank, you must save your scene, close Maya, reopen the file, and click the default button again.

Next click the "**QC settings**" tab and set the values according to your model's preferred attributes. The static prop checkbox should only be checked if you want the object to not be effected by physics. The prop will remain completely motionless (and will use fewer resources) of you check that box.

As for the mass and scale, the numbers range from 1-100 for mass, and are infinite for scale. These values can easily be modified later on by editing the QC file directly. Fill in the object's weight and size and the scale (the value is a multiplier).

The only remaining values to be set are your "**surface prop**" and your "**key value**". These are the materials that control what sounds, debris, and physics the source engine applies to in game interactions with your model.

Press "**FULL COMPILE**" at the bottom of the window, which brings step 2 to a close. Now, if you saw a small window with a blue loading bar pop up twice in a row, you successfully exported.

Step 3 – Compiling Your Exported Model

First, we must go to our exported files, which have been sent to our "**modelsrc**" directory. In the "**modelsrc**" directory you will see new files. Each file will be labeled with the Maya scene name and some will contain a suffix. The files with extensions **".smd**" are your models, one being the reference model, one being the physics model, and one being the idle animation.

Each of these is needed for the compiling process. You will also notice another file, your "QC" file. Open this file with Notepad and you will see something very similar to this:

```
$modelname"C:/ProgramFiles/Steam/steamapps/Your-User-
Name/sourcesdk_content/cstrike/modelsrc/Butte02.mdl"

$scale 2
$surfaceprop "rock"
$staticprop
$body "Body" "Butte02"
$sequence idle "Butte02_idle" loop fps 1
$collisionmodel "Butte02_phy.smd"
{
$mass 1
$concave
}
```

We need to edit this "QC" file because the exporter enters the wrong information. There are only a few changes to make to the file, some of which are your personal preference.

First off, the top line must be changed. The compiler we will use will interpret your QC file and place the necessary files in the specified locations. This first line is the directory that the compiler will place your finalized model. The compiler knows where the model directories are for the specified game, so the only directory information that you need to enter is for your own organizational preferences.

In this example, I have decided that I want my coconut model to be placed in the models directory inside a folder called "palm_tree" as the palm tree model will contain separate coconut models so that players can shoot them out of the palm trees. The folder is only there because I like to organize my models rather than pack the models directory full of random model files. As you can see below, the top line simply indicates that I want the compiled model to be placed in that folder. The compiler will create the directory for you if it does not exist.

```
$modelname "palm_tree/coconut.mdl"

$cdmaterials "models/palm_tree"
$scale 42
$surfaceprop "Wood_lowdensity"
$keyvalues { "prop_data" { "base" "Item.Small" } }
$body "Body" "coconut"
$sequence idle "coconut_idle" loop fps 1
$collisionmodel "coconut_phy.smd"
{
$mass 2
$concave
}
```

Optional Step:

You may notice that the Maya exporter has created an empty folder in your materials folder automatically that has the precise location you will be placing the finalized texture file; they are not important and can be removed.

Next, choose the precise location you will be placing the finalized texture file (NOT YOUR TARGA, THE VMT AND THE VTF). Looking at the example above, I added a line to the coconut one called "$cdmaterials".

*This will add a folder that can be used to organize your textures with in your "**materials**" folder. You do not need to do this, in fact, you may want to take full advantage of the fact that the Maya exporter already took the liberty of adding a directory with the same name as your scene file and place your textures in there. If not, just remember that they will stack up over time and you should remove them.*

Below this line there is a $scale command where you can modify the scale. The number can be a float. Once all has been changed and everything looks good, you may save the QC file and close it. Now we must navigate back to our "bin" directory and open the program you installed at the beginning of the tutorial called "GUIStudioMDL".

Editors' Note: GUIstudioMDL will want you to tell it where StudioMDL is located. Since the wonderful early November 07 updates, this has been moved to two separate bin folders under "ep1" and "OrangeBox" depending upon the desired game

GUIStudioMDL is very quick and easy. Select the game you are working with from the pull-down menu then open your QC file, "Compile". When the compiler has finished you may exit out of it.

Step 4 – Converting Your Texture Files

Begin with the program known as "**360g**". Now open "**360g**" (in your "**bin**" folder) and you should see icons that somewhat resemble trapper-keepers, the creator of this program chose to replace the common "manila folder" icon that we are so accustomed to with this new "black trapper-keeper" one.

Open directory the materials directory from our QC file. Simply drag and drop our **TARGA FILE DIRECTLY FROM** our "**materialsrc**" folder so that we see it appear in the window in thumbnail version. If you have a normal map to accompany it, drag this in as well.

Click file, and select the first option "New Materials File (VMT)…" and a window will pop up so we can see our options in defining how the Source engine will render our new texture file.

Since this texture will be used on a model, it is CRITICAL that we choose "**Vertex Lit Generic**" from the first pull-down menu entitled "**Shader**". The next pull-down menu is the texture's "**Surface**" material. In the Texture area under "**Base**", simply select the only texture you see listed and then select the normal map file under the "**Bump**" option.

Optional Note:

*Once this has been done, I recommend adding some contrast by selecting the "**Contrast**" checkbox under the "**Specular**" area to give it a more realistic look. If you happen to have an alpha channel on your texture map, then you will have to create two separate texture files. One containing the portions of the model with alphas, and another without them, you must also go back to step two and into Maya again so that you can change the UV's to work as such.*

Open the TARGA version of the normal map in Photoshop and overwrite the one that 360g automatically created in the directory you specified. When saving a normal map VTF from Photoshop, you will notice a new and unfamiliar window pop up before you overwrite the file. This is the VTF modifier window, which allows you to specify specifics about the file's format. But all you need to do is pull-down the "**template**" menu and find "**Normal Map**". 360g is notorious for scrambling normal maps when it converts them into VTF format. If you happen to notice that your texture map is also messed up, repeat the process for the color map as well and select default from the template menu.

There is one final step before you are done with this process in the VMT file. But this time, you need to open it with Notepad. Find that texture directory and open the VMT file for your model's texture. Once here, the file should look like this:

```
"vertexlitgeneric" {
   "$basetexture" "models\banyan_tree\Banyan_Color"
   "$bumpmap" "models\banyan_tree\Banyan_Normal"
   "$envmap" "env_cubemap"
   "$envmapcontrast" 1.000000
   "$envmaptint" "[ 0.5020 0.5020 0.5020 ]"
   "$surfaceprop" "wood"
}
```

All you have to do is delete the line that reads "$envmap env_cubemap" unless you want the material to reflect light much like a steel counter-top. Save the file and exit notepad.

To see your sweet new model, run the "Source SDK" and select the "Model Viewer" from the list and open it.