# 3do2Scm

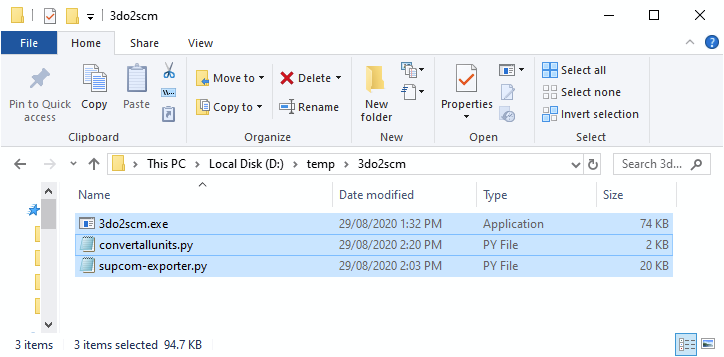
# A workflow to import Total Annihilation models to Supreme Commander

There are a couple of [blender plugins](https://github.com/Oygron/SupCom_Import_Export_Blender) that allow import/export of SupCom SCM files. But the workflow to get your old Total Annihilation 3do files into blender with textures and correctly oriented bones/armatures etc in the first place is ... well I couldn't find an easy path.

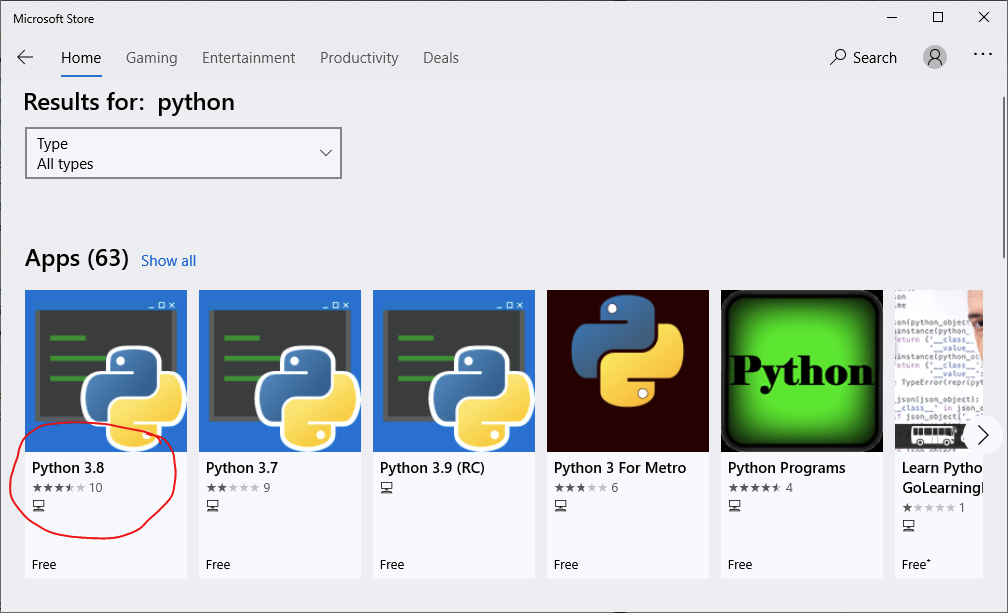
So I rolled up my sleeves and pumped out this tool to conver from 3do directly to scm. Apologies for the crudeness of the solution and the manual steps to convert the texture formats at the end but I didn't want to spend a heap of time on this.

## Instructions

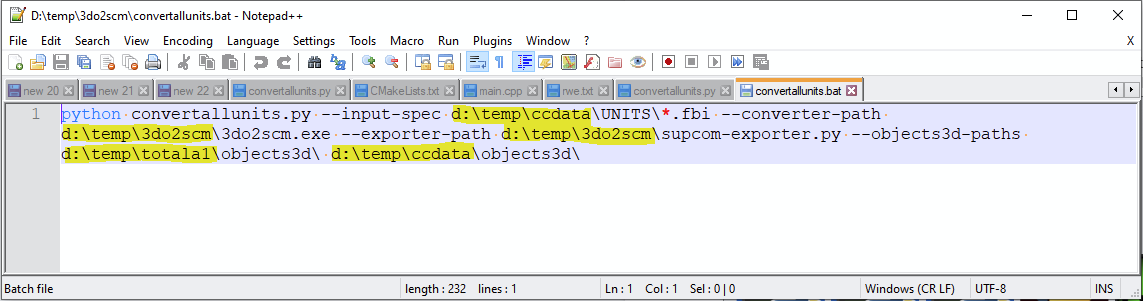
Use [HPIView](http://units.tauniverse.com/tutorials/tadesign/tadesign/ta-files.htm) to extract Total Annihilation data somewhere. In the following I’ve extracted mine to d:\temp\total1 and d:\temp\ccdata\. We’re going to be using the “units” and “objects3d” subdirectories directories.

Extract contents of release to someplace. I’ll use d:\temp\3do2scm:  


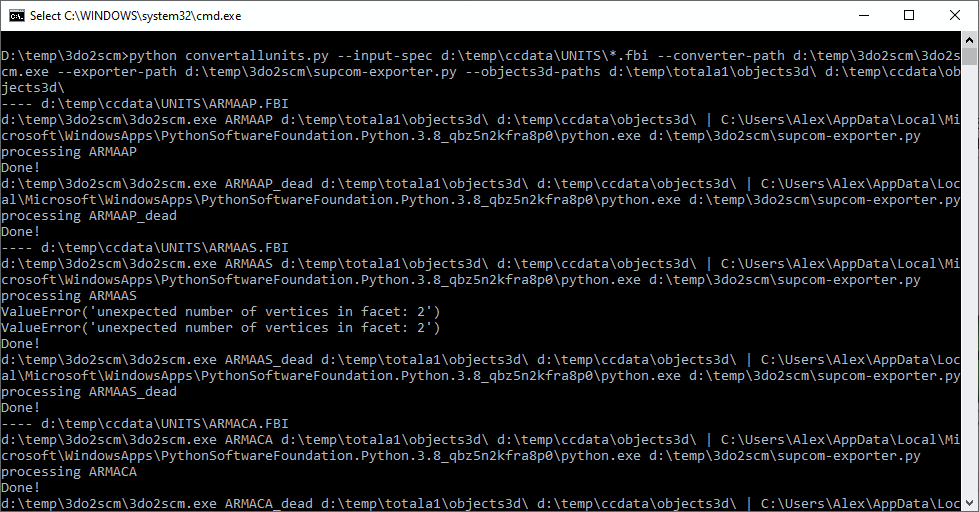
Install some version of python3. You can find it in the Microsoft Store, or install [anaconda](https://www.anaconda.com/products/individual). It may work with python2, give it a go if you don’t have python3 installed but do have python2 available.



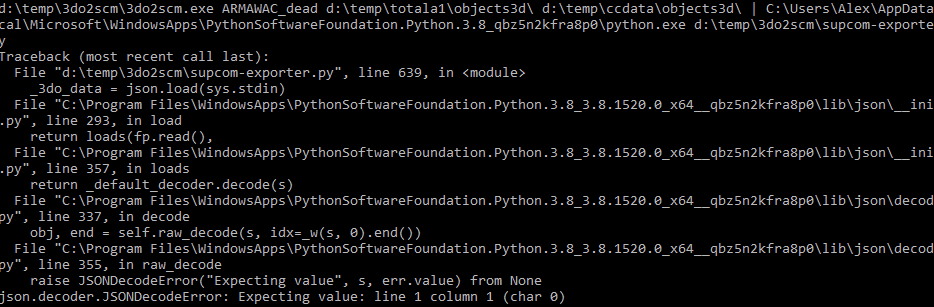
Edit the file “convertallunits.bat” and change the paths to match where you extracted everything



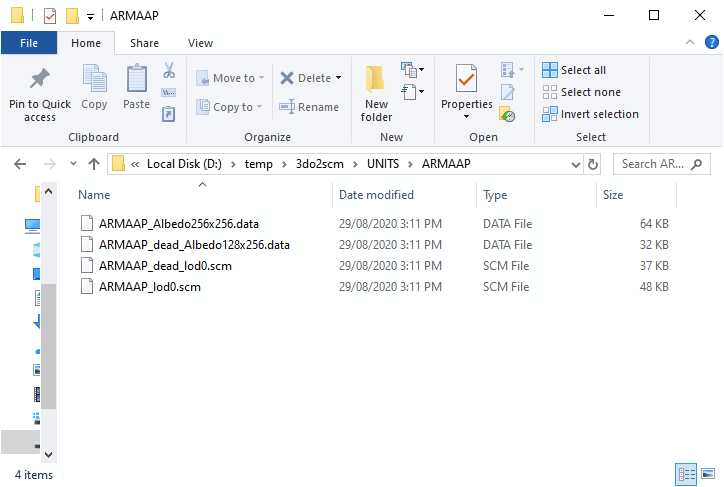
Double click on “convertallunits.bat” and observe the output. It should take a few minutes.



You may see the odd message about “unexpected number of vertices”. Seems to be harmless. And the odd json decoding error about “expecting value: line 1 column 1”. That just means there is no “\_dead” model associated with the unit.



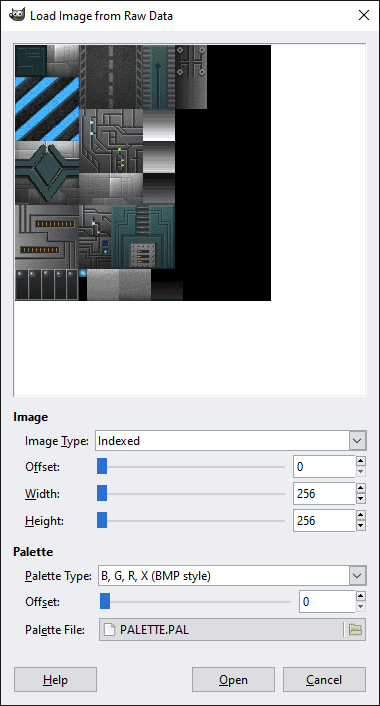
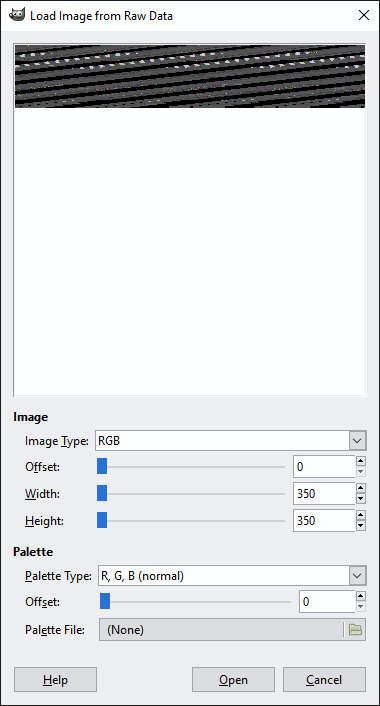
If all goes well you’ll now have a new directory called “UNITS” in your d:\temp\3do2scm directory:



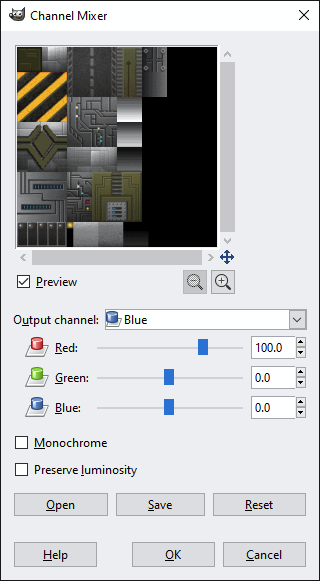
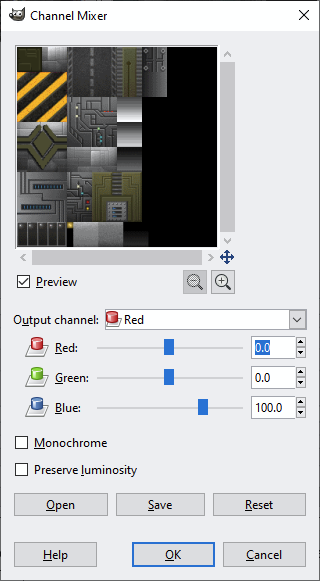
The .scm files contain the 3d model. And the .data files contain the texture images in raw format. Now we need to convert the texture images to DDS format so SupCom engine can load them. It may be possible to do them all at once using [imagemagick](https://imagemagick.org/index.php) or some other tool. However I’ve done mine manually using [gimp](https://www.gimp.org/). Read on.

Open Gimp. Drag at “.data” file into gimp. You should get the “Load Image from Raw Data” dialog (left figure below). Enter some data into the dialog to format the image correctly (right figure below). The “PALETTE.PAL” file comes from the Total Annihilation game files: specifically d:\temp\totala1\palettes\PALETTE.PAL

Once you see some sensible structure in the image, click “open”. Don’t worry about the inverted colours (blue stripes instead of yellow) for now. We’ll fix that next.



To correct the inverted colours: go “image”, “mode”, “rgb”. Then “colors”, “components”, “channel mixer”. Set the red channel as 100% blue and the blue channel as 100% red as pictured below. Leave green as 100% green. Alternatively, if you know how, modify the PALETTE.PAL to contain the correct colours to begin with and save you some time in the future.



Then go “file” “export” and export as a png. Then use [DDS Converter](https://vvvv.org/contribution/dds-converter) or similar to convert the png into a DDS file and rename to <unitname>\_Albedo.dds.

Wash rinse repeat for the \_dead.data (if it exists) to create <unitname>\_dead\_Albedo.dds.

Unfortunately the creation of the team colours (<unitname>\_SpecTeam.dds) isn’t supported so you’ll have to do that manually. Open the (non-dead) png again in gimp. Use your intuition and creativity to work out which tiles should be team coloured. Or just copy another unit’s SpecTeam to get some random mapping which will at best look mediocre, but at least provide you with an idea of what you’re aiming for.

You now have:

* <unitname>\_lod0.scm
* <unitname>\_Albedo.dds
* <unitname>\_SpecTeam.dds
* <unitname>\_dead\_lod0.scm
* <unitname>\_dead\_Albedo.dds

This should be the minimum required to create the 3d model side of the unit in the SupCom engine.

The remaining files to actually create the unit are:

* <unitname>\_dead\_prop.bp
* <unitname>\_heap\_prop.bp
* <unitname>\_unit.bp
* <unitname>\_script.lua

Which this tool unfortunately can’t help you with, but they’re text format and there are plenty of examples to look to for reference.

Any questions / feedback, leave me a message on the FAF forums.

Good luck!