# 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.

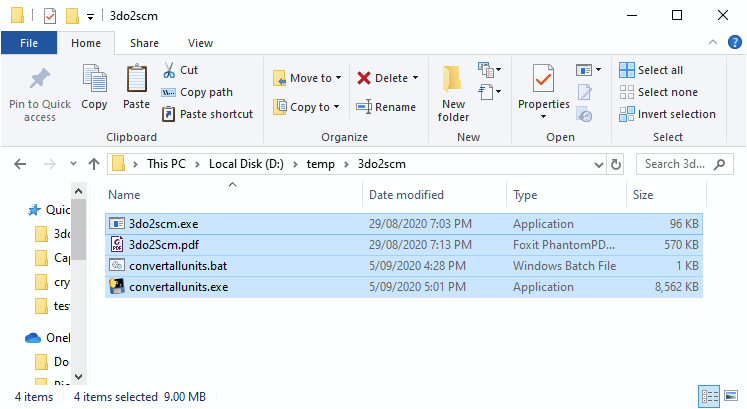
So I rolled up my sleeves and pumped out this tool to convert from 3do directly to scm. It makes heavy use of some fine code for reading Total Annihilation data structures by Michael Heasell from [here](https://github.com/MHeasell/rwe) as well as the [blender plugins](https://github.com/Oygron/SupCom_Import_Export_Blender) themselves by Dan, Brent, GeomanNL, Darius, Oygron, so big shout outs to then all is due.

And since the 3do/SCM is only half the story, we include a tool to assist in converting TA animation scripts from their native BOS/COB format into SupCom’s SCA format. We’ll cover that in the instructions too.

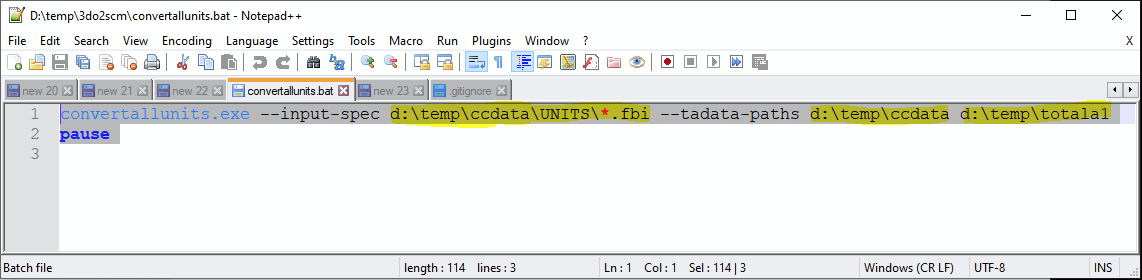
Apologies for the crudeness of the solution and the manual steps to convert the texture formats at the end but I didn't want to spend a heap of time on this.

## Model Conversion (3do to SCM)

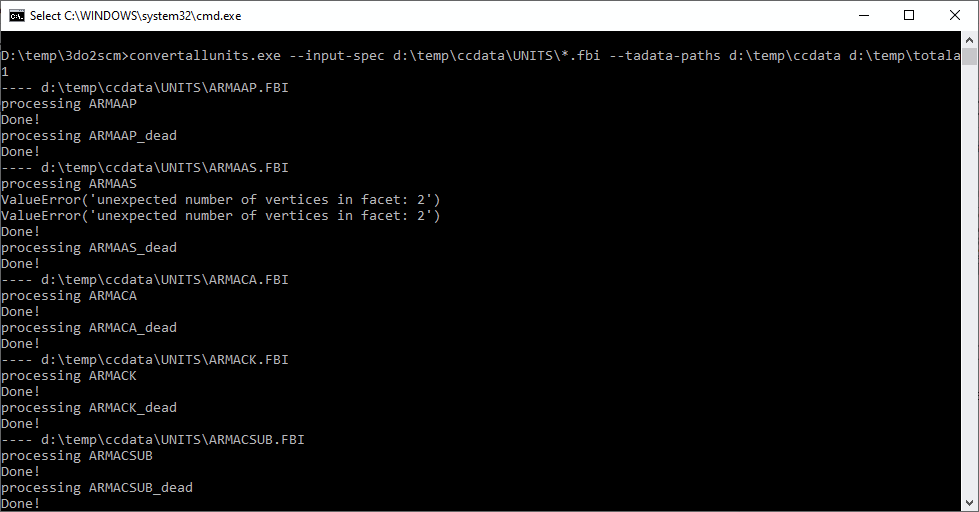
Use [HPIView](http://units.tauniverse.com/tutorials/tadesign/tadesign/ta-files.htm) to extract Total Annihilation data somewhere. In the following I’ve extracted mine to d:\temp\total1 and d:\temp\ccdata\. We’re going to be using the “units” and “objects3d” subdirectories directories.

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


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

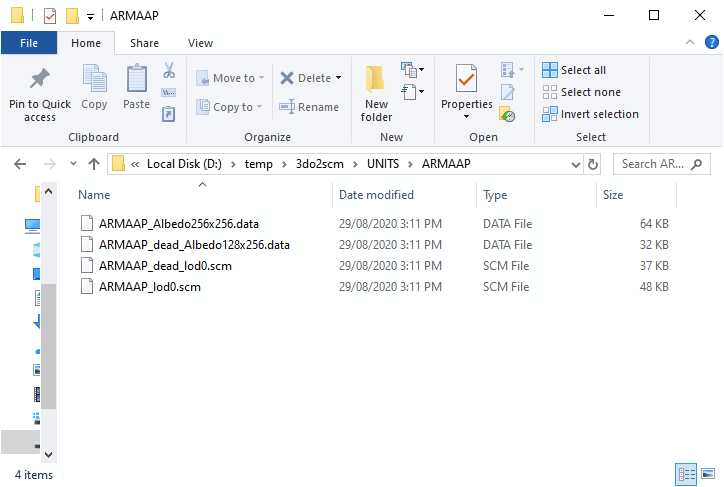


Double click on “convertallunits.bat” and observe the output. It should take a few minutes.



You may see the odd message about “unexpected number of vertices”. Seems to be harmless.

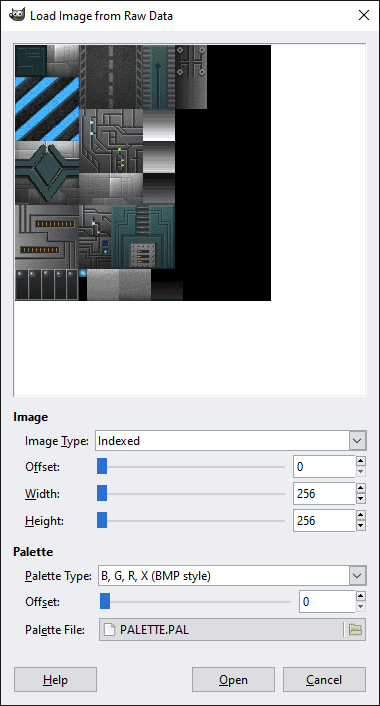
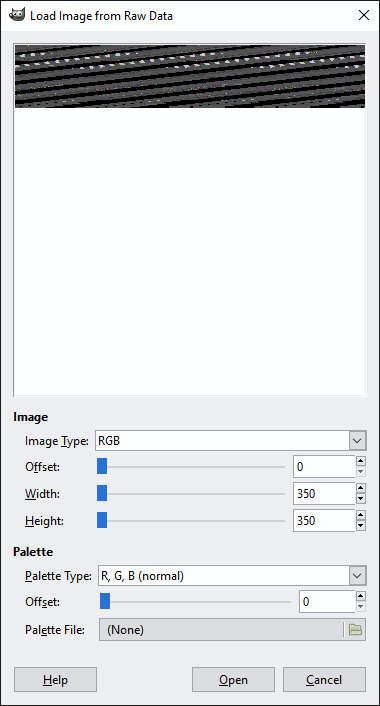
If all goes well you’ll now have a new directory called “UNITS” in your d:\temp\3do2scm directory:



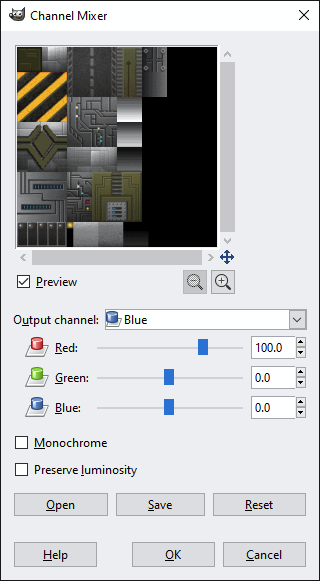
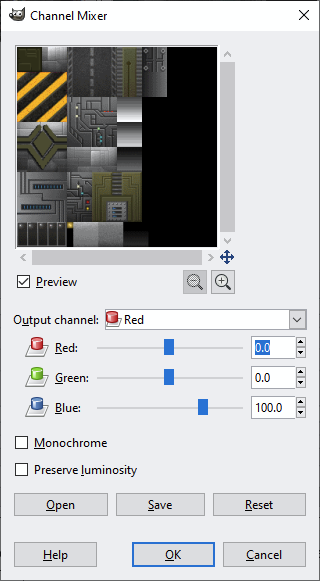
The .scm files contain the 3d model. And the .data files contain the texture images in raw format. Now we need to convert the texture images to DDS format so SupCom engine can load them. It may be possible to do them all at once using [imagemagick](https://imagemagick.org/index.php) or some other tool. 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](https://forum.faforever.com/topic/128/convert-total-annihilation-3do-to-supcom-scm).

Good luck!

# Animation Conversion (BOS/COB to SCA)

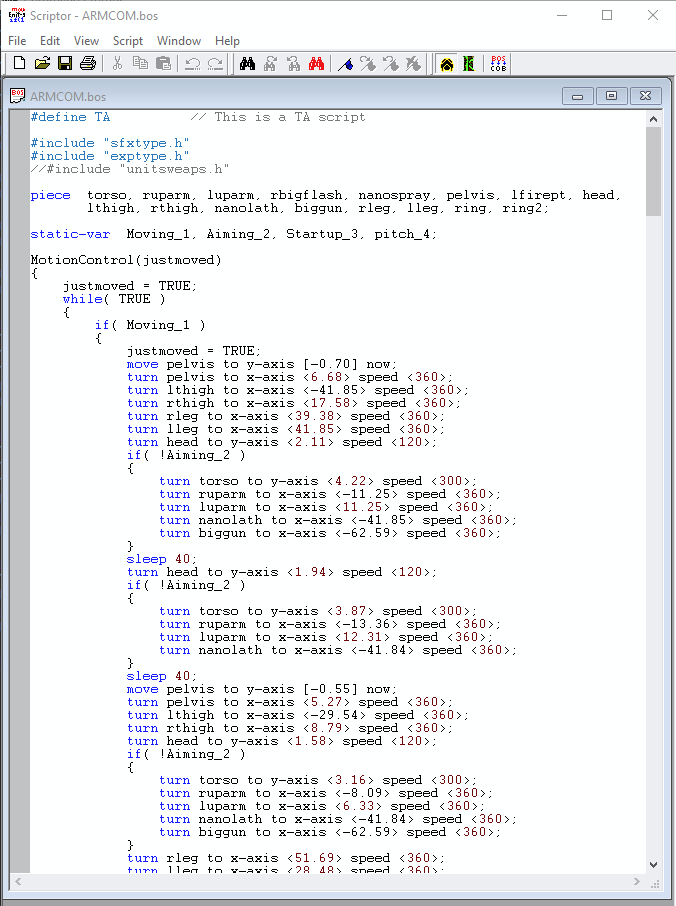
TA animation scripts (BOS files) are compiled (to COB files) and run in a virtual machine with loops and condition statements and local variables and generally allow for the animation to be tightly integrated into the TA simulation, whereas SupCom SCA animation files are pre-compiled sequences of movements without any decision points in them. Instead the SCA animations are triggered from a LUA script, which would do all the integration with the simulation environment.

Therefore a seamless conversion of TA’s BOS/COB files to SupCom LUA/SCA may be extremely complex (or a fun challenge?). But in the interests of pragmatism we’ve instead opted for operator assist rather than full automation.

The approach here is going to be to copy/paste snippets of BOS code for compilation into SCA files. It’s up to the user to find the snippets of interest and to activate the SCA animations from LUA code. We’re going to call the snippets of BOS code nBOS for “not-BOS” since the language is going to be greatly simplified and a few special keywords added.

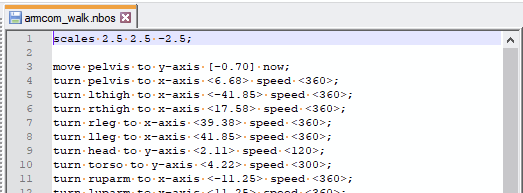
Lets walk through the process of obtaining a snippet of BOS code for making the Arm Commander walk, and compiling it into an SCA file.

First, grab a copy of [Scriptor](http://switeck.tauniverse.com/). Run it and open a .BOS file or a .COB file from your TA data directory. I’m going to load ARMCOM.bos from the Total Mayhem mod.

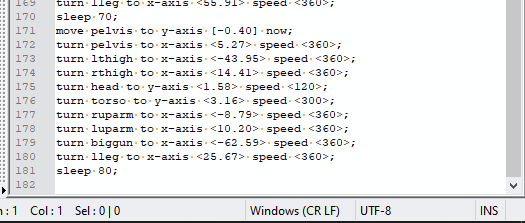


Look for the MotionControl() function. It contains a whole cadenza of “move” and “turn” and “sleep” statements enclosed in while loops and if statements. Let’s copy/paste all the move/turn/sleep statements from the blocks where the ARMCOM is just “moving” but not “aiming” (ie !aiming. “!” means “not”). Save them in a new file using your favourite text editor (eg notepad++) and call it “armcom\_walk.nbos”.

Add a special “scales” statement to the top to indicate the TA-to-SupCom x,y,z scale factors to use. Use 2.5,2.5,-2.5 to match the output of 3do2scm. Or use 1,1,1 if you’ve written a script from scratch using SupCom coordinates.

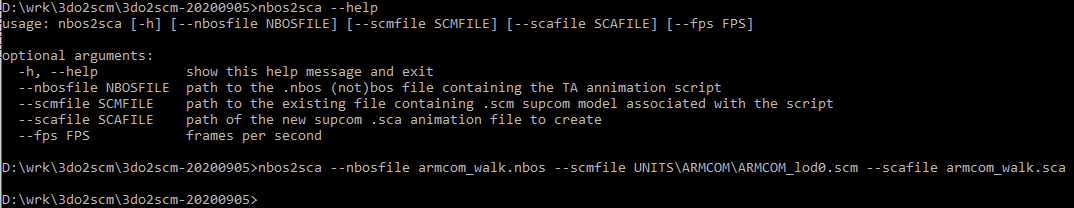


…



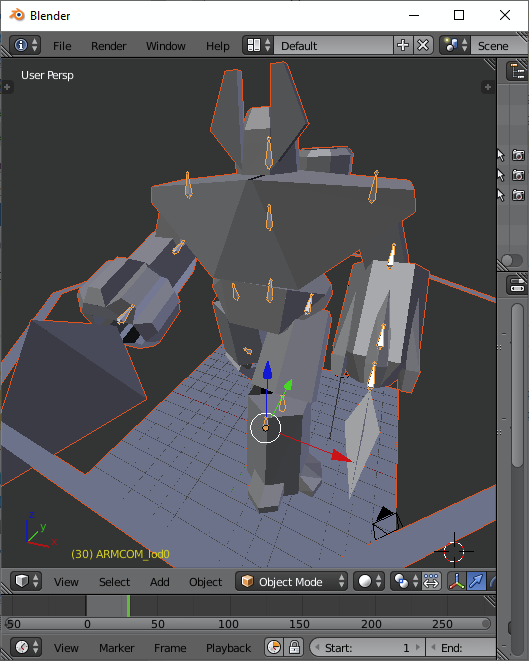
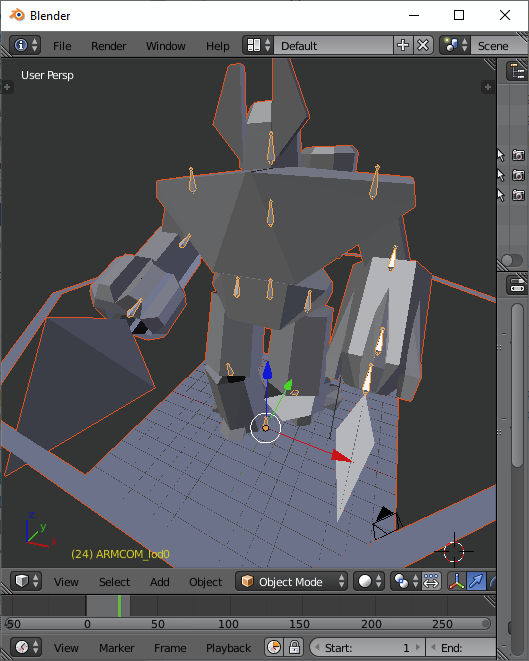
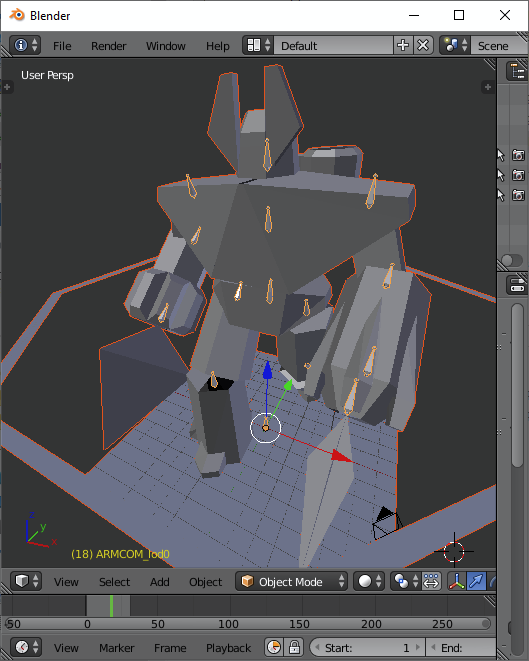
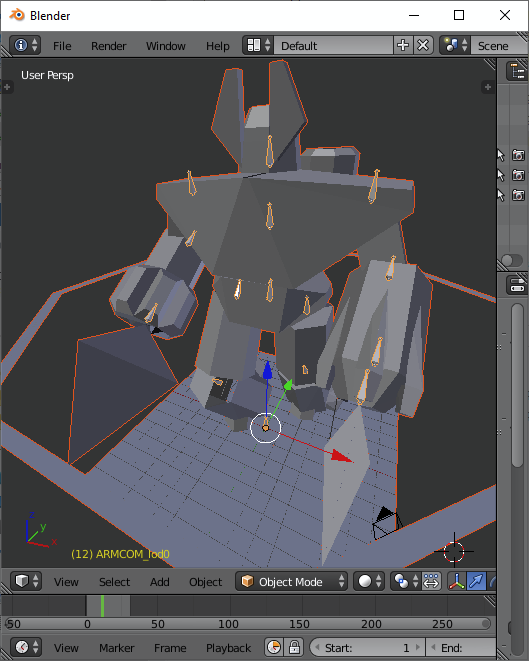
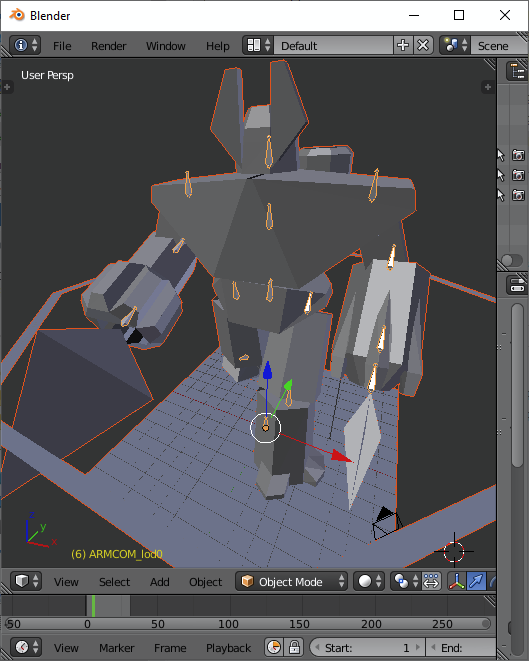
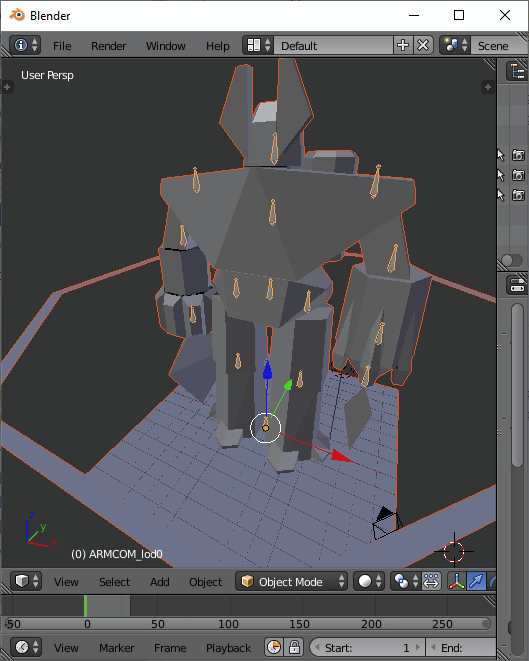
Note that the original BOS file has a list of “pieces”. Our nbos2sca converter tracks the poses of all bones in the model regardless of whether they’re mentioned in the script. As an animator newbie, I’m unclear as to whether or not this is useful / required. Please [let me know](https://forum.faforever.com/topic/128/convert-total-annihilation-3do-to-supcom-scm) if the tool needs a feature to restrict the animation data to only a subset of listed bones.

Run the nbos compiler (open command prompt and execute as below; or create a .bat file with the nbos2sca statement in it and double click on it):



(ARMCOM\_lod0.scm is the output of 3do2scm run on the Total Mayhem version of ARMCOM.3do).

You should now have a shiny new .SCA file. Let’s check it out in [Blender](https://www.blender.org/). Use the [Blender plugins](https://github.com/Exotic-Retard/SupCom_Import_Export_Blender) to load the .SCM model and then again to load the .SCA animation.



That’s it. I hope you’re able to make some gains from this tool.