

Converting FMS models for ClearView

Converting Planes

Guide and tutorial to aid new modellers wishing to convert existing models created for FMS to the ClearView flight simulator. This guide covers basic conversion techniques to create a model that will fly and have animated control surfaces.

Converting FMS planes for ClearView

Contents

What you need.....	1
Preparing to convert	2
Step 1 – The import.....	3
Step 2 – Tidying up the import.....	4
Step 3 – Sorting out textures	8
Step 4 – Adding control/collision points.....	13
Step 5 – Params.txt.....	16
Step 6 – Engine Sounds.....	17
Step 7 – Adding control surfaces and props.....	20
Preparing.....	20
Cutting out the ailerons	24
Cutting out the elevator	27
Cutting out the rudder.....	27
Cutting out the tail wheel	29
Animate the rudder	31
Creating an animated propeller	33
Some final tidying up	35
Step 8 – Packaging.....	36

Converting FMS planes for ClearView

What you need

You will need ClearView 4.22 or later installed in order to convert FMS (or .X) models, at the time of writing ClearView is up to version 4.35.

You don't need to have FMS installed on your computer but sometimes it can be helpful to check that a model looks as you want it to before trying to convert it. Once you start the conversion process it can also be useful to take a look at how it is supposed to appear.

You will also require AC3D. This can be obtained from <http://www.ac3d.org>

An image conversion program to change bitmaps (bmp files) into JPG files. Paint that is built into MS XP can do this though you may want something more sophisticated if you want to modify the paint schemes yourself.

Converting FMS planes for ClearView

Preparing to convert

For this example I am using the Watt Age Decathlon ARTF model from Gary Gunnerson's FMS Models website

http://gunnerson.homestead.com/files/fms_models.htm

Start by extracting the files from their archive. It doesn't really matter where you do this though if you have the FMS putting a copy there can be useful.

Ensure that the files include a .X file. This is the file that ClearView will use to convert into its own format. This particular model contains seven files:

- W-decathlon.par – this is the FMS flight parameters file.
- W-decathlon.x – this is the 3D model file in an FMS format.
- W-decathlon_prv.jpg – this is the thumbnail that FMS displays in its open model dialog.
- W-decathlon.bmp – this is the texture file used to 'cover' the finished model.
- W-decathlon.wav – this is the engine sound file
- W-decathlon.mqo – many of the FMS planes are built using a freely available 3D editor called Metasequoia. The .mqo file is its native storage name.
- W-decathlon_readme.txt – is a readme that usually contains amongst other information the name of the original author and any subsequent conversion authors.

Having mentioned the readme it is always a very good idea to check this for the email addresses of the original author and anyone who has modified the models in the past.

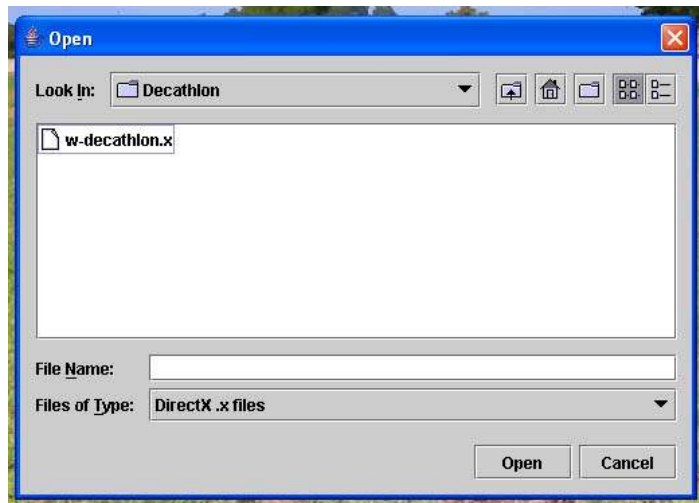
It is good practice to email the people named to check if they have any objection to your conversion. Technically it is a copyright issue. Most people don't mind you converting their models as long as you are not copying them from a commercial product or intending to make money from selling the conversion, but it is still best to ask first. This is particularly important if you intend to distribute your conversion.

It is also normal and good practice to name them in the final readme but more about that later.

Converting FMS planes for ClearView

Step 1 – The import

To start the process of conversion start ClearView and when it has finished loading select “Import Model” from the “Settings” menu. This will display an ‘Open’ dialog box. Navigate to the directory in which you have stored the model for conversion and select the .X file.

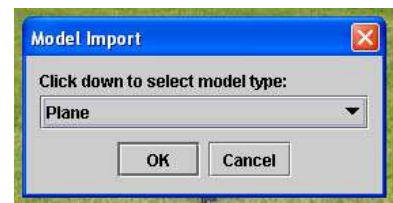


ClearView will then ask for the model name. This is both the name of the model as it will appear on the planes menu as well as the directory name it will store the files under. Type the name exactly as you want it to appear. In this case I will leave it at the default.

It may be obvious but ClearView can't tell the difference between a helicopter and a plane so you need to tell it.

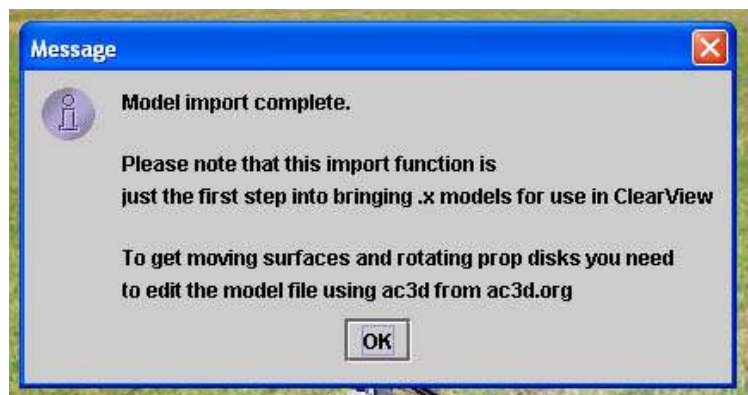


Change it to plane and let it do its stuff.



When it's done it displays this message. As it says this is just the start.

Actually at this point it may be flyable in ClearView. If you wish to try you can select it from the planes menu.



It is likely that it will be a funny colour, may be displayed at a funny angle and might be continuously crashing. These are all things we are now going to fix.

Converting FMS planes for ClearView

Step 2 – Tidying up the import

The conversion process results in a lot of duplicate information in the model file that has to be removed and some modification made to help both with texturing and general model presentation.

The first step is to load the model into AC3D. ClearView converts the FMS model file in one called body.ac in a directory associated with the name you provided in step 1.

In this case it can be found in:

‘SVKSystems/ClearView/models/w-decathlon/data/body.ac’.

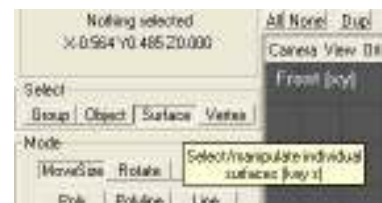
Once the model is loaded if you take a look at the model information by pressing F6 you will see that it consists of:

- 6 meshes
- 7 objects
- 1158 surfaces
- 6888 vertices

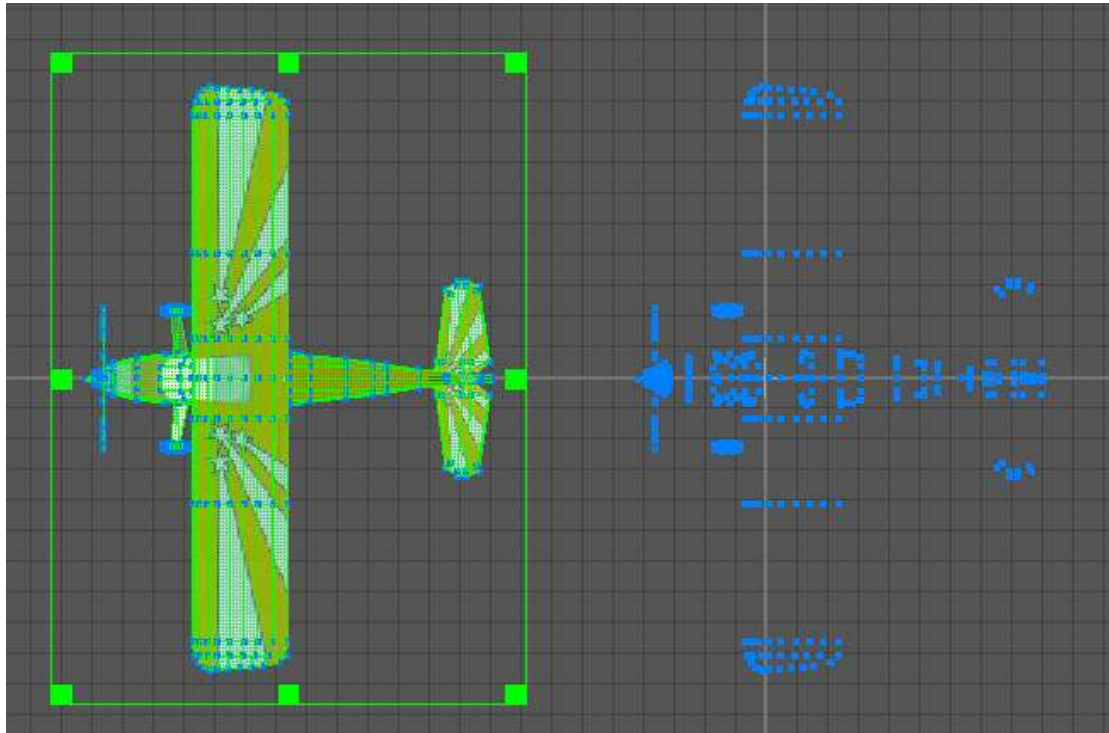
This is quite clean for a conversion many can have 30,000 or more vertices depending upon how it was originally drawn. Since each surface should have either three or four vertices you can see that there are more vertices than necessary. The first stage is to remove the un-necessary ones.

To remove un-necessary vertices first close the model information and then ensure that nothing is selected. (Click somewhere outside the bounding box.)

Switch to surface mode and then select any surface on the model. Then select all surfaces using CTRL-A (or edit->select all). In one of the ‘orth’ windows you can now drag the plane to one side and it will leave behind a number of vertices in the shape of the model.



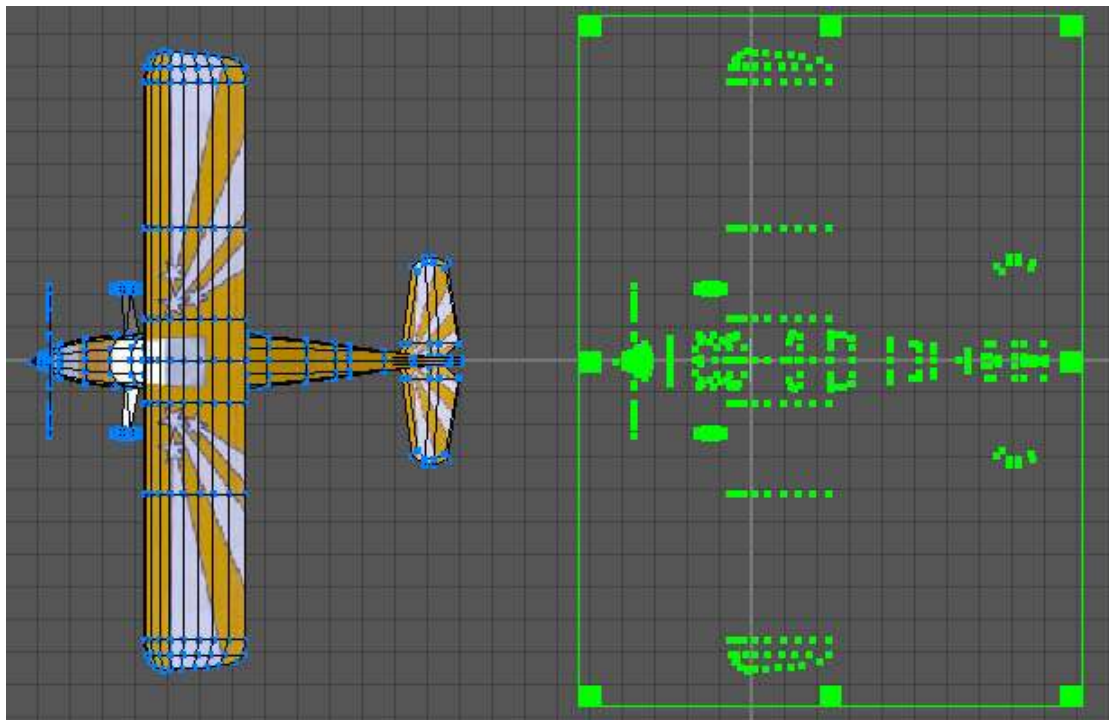
Converting FMS planes for ClearView



Now switch to vertex mode:



Select all the unwanted vertices on the right:



And delete them, leaving just the model.

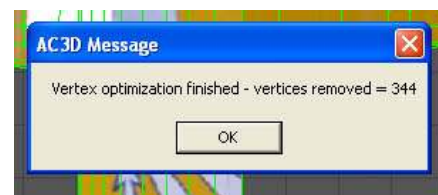
Converting FMS planes for ClearView

If you now look at the model info again (F6) you can see the difference. There are now:

- 6 meshes
- 7 objects
- 1158 surfaces
- 1291 vertices

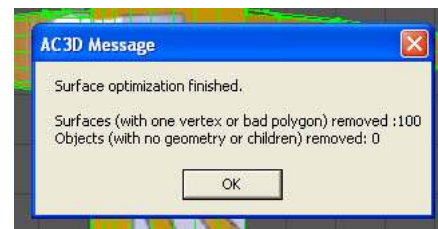
At this point it is worth saving your progress. Then switching to object mode select everything and then drag the model back towards the centre as it is easier to work on in this position.

Although the majority of the unwanted vertices have been removed there is still likely to be some more left. It is now time to optimise the vertices. Make sure everything is selected and then optimise the vertices by selecting object->optimise vertices. It will tell you when it is complete and how many, if any, unwanted vertices it removed.



Incidentally this process would not have removed all the unwanted vertices that were deleted earlier as this only works for vertices that are part of a surface, the ones earlier were not. That is why they were left behind when we moved all the surfaces to one side.

At this time you can also optimise surfaces to remove any bad surface. A bad surface is one that has less than three vertices. To optimise surfaces again select everything in object mode then select object->optimise surfaces. It will tell you when it is complete and how many, if any, bad surfaces it removed.



The basic tidying up is now almost complete. Two other checks are worth doing at this point. Checking for invalid polygons and checking for any surface made up of more than four vertices.

Select surface mode and ensure that nothing is selected. Now chose edit->select surfaces->invalid poly. This will select any surface that is an invalid polygon. In this case there isn't any but for some models there will be some selected. If found they can either be individually fixed or just deleted. Sometimes deleting invalid polygons leaves holes in the model that will need to be patched but generally the changes are invisible.

To check for surfaces made up of more than four vertices (ClearView will refuse to display such surfaces) again make sure nothing is selected then chose edit->select surfaces->> 4 vertices. In this case nothing is selected but if found they can very easily be fixed. Whilst the surfaces are still selected chose surface->triangulate and it

Converting FMS planes for ClearView

will reduce the problem surfaces to triangles which are valid. This may have some effect on the look of the model but generally this can be fixed later.

This part of the tidy up is complete so save the file. This is important as modifying textures can often result in the need to try different approaches and sometimes it is easier to start again rather than try to fix a change.

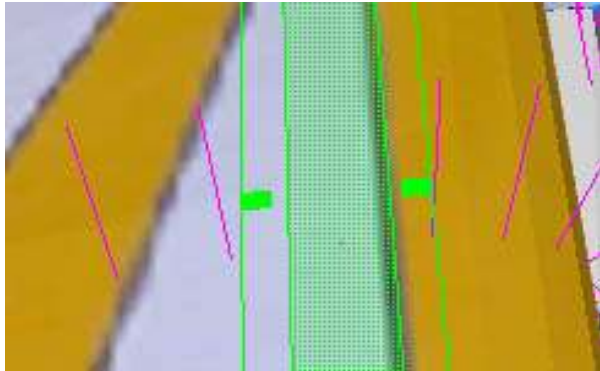
With experience you would probably merge objects, check and modify surface characteristics at this stage to improve the final look of the model. However, that is for an advanced guide.

Converting FMS planes for ClearView

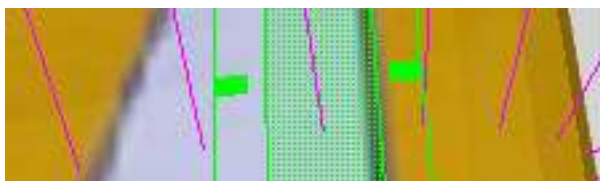
Step 3 – Sorting out textures

Generally it is a good idea to check ‘normals’ at this point. A ‘normal’ is the direction a light ray would travel if it is ejected at right angles to a surface. The ray is ejected from the front surface only. You can see ‘normals’ by turning them on in the ‘orth’ and 3D windows. Select orth->normals and 3D->normals to see them. This will make the model look somewhat hairy. A tip here is to change the length of the displayed ‘normals’ as it make them easier to see in complex models. The length can be changed in the settings menu (file->settings) under the appearance tab. Change “Size of display normals” to 0.075 or to your liking.

A surface that doesn’t display a ‘normal’ is set so that it has its back face towards you. Generally this is wrong and in ClearView usually makes that particular surface black or very dark. It certainly looks odd. Although this model doesn’t have a problem with inverted ‘normals’ one has been purposely inverted in the picture below to show the effect.



If you look along the line of pink ‘normals’ you can see the dot in the middle of the green area. This is an inverted ‘normal’ with the rest being correct. This would give rise to bad shading in ClearView. To fix a bad ‘normal’ select the surface (use surface mode) as in the illustration then chose surface->flip normal to make it correct.



Sometimes it can be difficult to see wrong ‘normals’ because the top and bottom surface of a wing can both be inverted. This means from the top you are actually seeing the ‘normal’ for the lower surface and vice-versa. A tip here is either to make the surface one sided so you can see missing surfaces or just flip the ‘normal’ for a surface and see if it gets longer. If it does get longer than that is the correct way round.

If you were to try looking at your conversion in ClearView at this point you would notice two things. The colours are strange and any writing is mirrored. The first is caused by the way ClearView loads bitmaps and the second by the different way in which FMS and ClearView use coordinates.

Converting FMS planes for ClearView

The first step to fix the colours is to convert the bitmap file to a jpeg file. This can be done with almost any image editor. In paint it is simply a matter of opening the file and then saving it as a .jpg file. Usually you will keep the same filename. Note however the file formats are different between a .bmp and a .jpg so it is not enough to just change the name.

In this case open the w-decathlon.bmp file in Paint and then save it as type JPEG (w-decathlon.jpg).

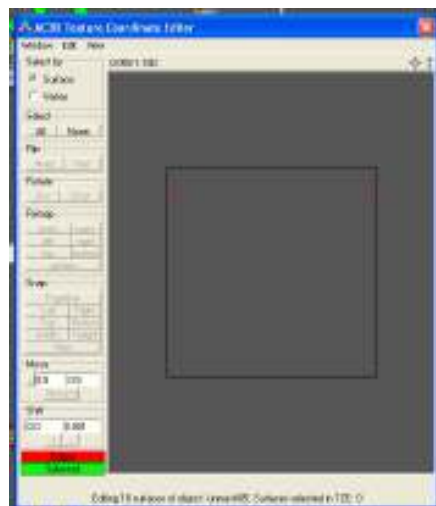
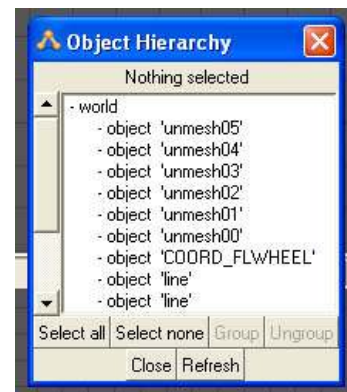
Now applying the change of texture file can be tricky as surfaces are often coloured using an attribute rather than a texture file. Applying a texture file inappropriately loses the colour and applies a texture that you don't want. So where possible, until you get skilled at textures and meshes, try not to change too much.

The first part of amending the textures is to generally change the texture from the old .BMP file to the new .JPG file. You cannot do this on a wholesale basis as you will assign textures to surface that shouldn't have one. The change has to be made on an object by object basis.

The simplest way to do this is using the object hierarchy display. This can be brought up either by pressing F8 or from Tools->Hierarchy View.

You should also be in object mode within AC3D to enable whole objects to be selected.

Each object in the object hierarchy view is an independent part of the plane. Because of the different ways in which models are created and in which ClearView converts them there may be no apparent reason why some elements are individual objects. However, changing that is an advanced topic.



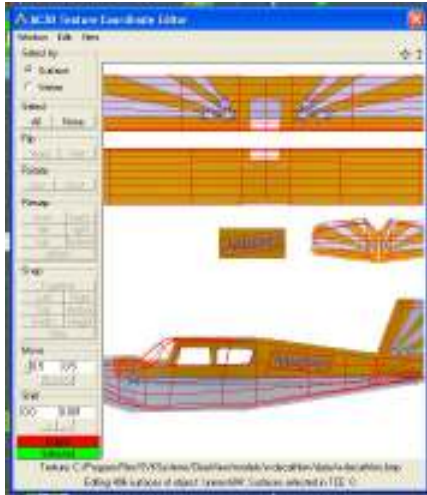
For now push the 'Select none' button in the hierarchy window to ensure nothing is selected then click on the first object, in this case 'unmesh05'. This will select, and highlight, that single object (in this case the two wing tips).

Unfortunately AC3D does not easily provide texture file information so we have to approach this from another angle. To find if the object has an associated texture and which texture file it is, it is necessary to bring up the texture editor by pressing F10 or Tools->Texture Coordinate Editor.

In this case the texture editor shows no background image and this means that this object does not have an associated texture file. We do not want to assign a file either. Close the texture editor.

Back in the Object Hierarchy view deselect object 'unmesh05' and select the next object 'unmesh04'. Again display the texture coordinate editor.

Converting FMS planes for ClearView



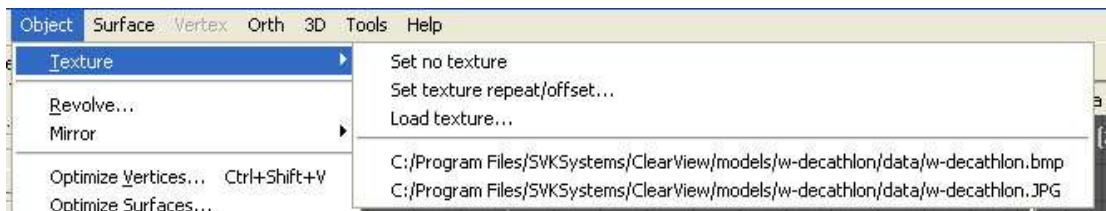
This time you can see that a background image is displayed. This is a texture file. Object 'unmesh04' has an associated texture. Incidentally, in models where there is more than one texture file you can see which one is being used as the filename is displayed at the bottom of the Texture Coordinate Editor window. In this case you will be able to see that it is w-decathlon.bmp.

It is now necessary to change this from the .bmp to the .jpg file.

To change the texture file, exit from the Texture Coordinate Editor. Then select the menu item Object->Texture->Load Texture. Navigate to the w-decathlon/data directory and select the w-decathlon.jpg file. This has now substituted the .jpg file for the .bmp file for this particular object.

If you now take another look at the Texture Coordinate Editor you will see that the texture is now being read from the .jpg file. No changes are needed to the coordinates because both files, being copies, are based on exactly the same coordinates.

Now repeat this process for the remaining objects. For subsequent objects that need to have the texture file changed it can be selected directly from the Object->Texture submenu rather than having to navigate to it through the Load Texture menu item.



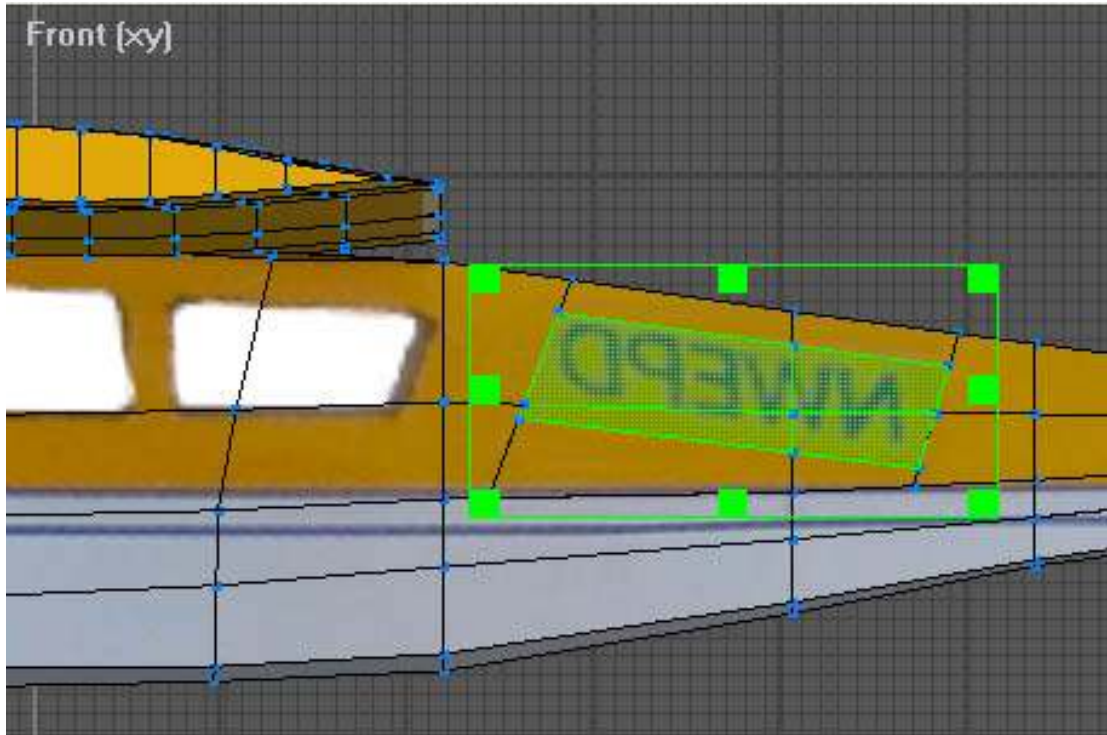
In this model's case no other objects have a texture.

Now we need to change the lettering to be the correct way round.

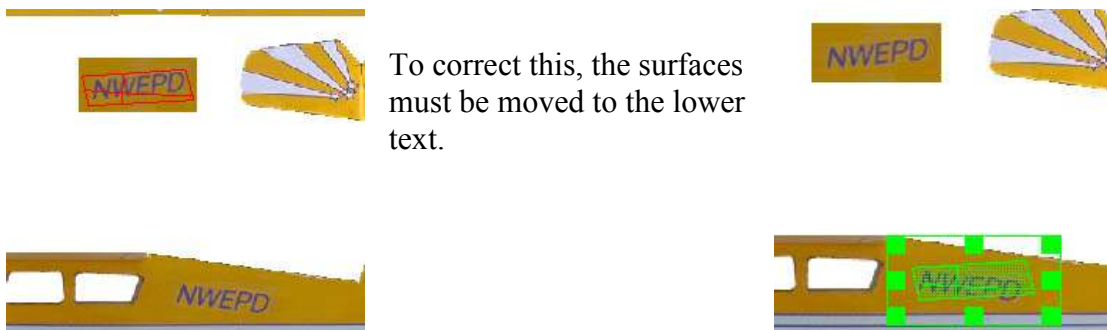
The author of this model has been very helpful in that the writing has been separated into their own surfaces making changing them around very easy.

Change to surface mode and in the front window select the four surfaces that make up the plane registration letters.

Converting FMS planes for ClearView



Now open the Texture Editor (F10 or Tools->Texture Coordinate Editor) and you will see that the surfaces are apparently mapped correctly but closer examination of the picture will show that although this is the left hand side of the aircraft it is in fact mapped to the right hand side registration letters.

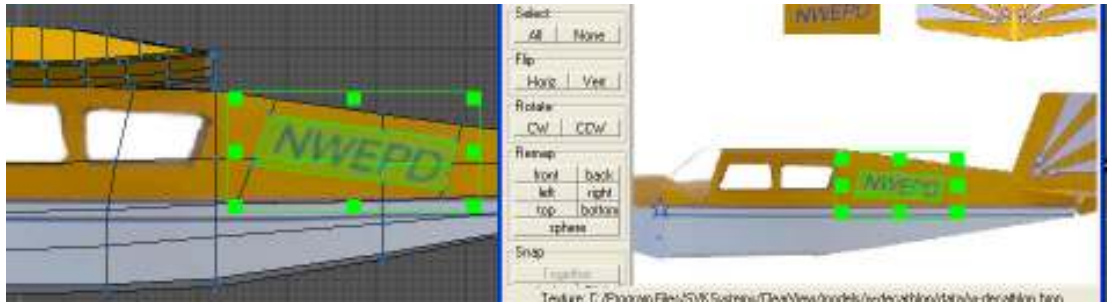


To correct this, the surfaces must be moved to the lower text.

Now the surfaces need to be adjusted to fit. If you take a look at the front window you will also see that the text is still reversed. Both problems can be most easily fixed by flipping the surfaces horizontally.

In the texture editor select all surfaces and then select 'Flip Horizontal' It is then just a simple matter of fine tuning the position until it encloses all the text and the appearance in the front window matches the desired look.

Converting FMS planes for ClearView



Close the texture editor window and then repeat for the right hand side of the aircraft. A tip here is to change the front window into a back window by selecting it from the Camera menu.

It is a good idea to save the model again at this point. You can load this model in ClearView and it will be displayed correctly. However, there is still work to be done. It is likely that ClearView will still be repeatedly crashing the model or displaying it in funny positions. This is because the model is missing some important reference points which we will add next.

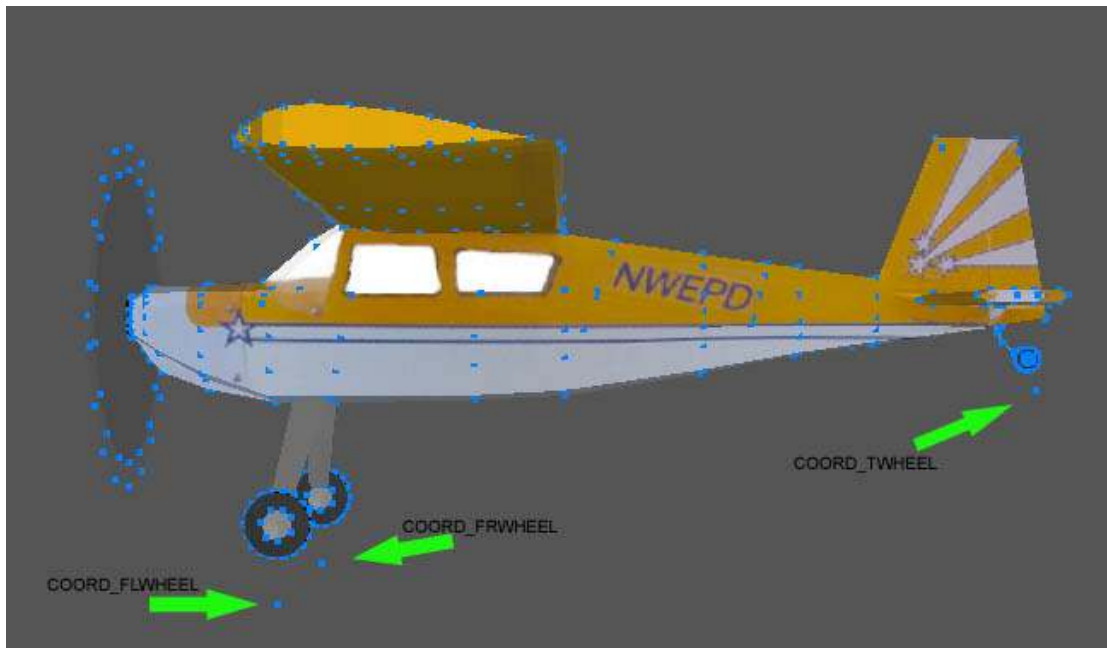
Converting FMS planes for ClearView

Step 4 – Adding control/collision points

ClearView uses some important reference points that describe the extremes of the aircraft and which it uses to understand where the model is in three dimensional space. These take the form of coordinate objects with specific names. These are:

- COORD_NOSETIP
- COORD_TALEND
- COORD_TAILTOP
- COORD_FLWHEEL
- COORD_FRWHEEL
- COORD_TWHEEL
- COORD_LWINGTIP
- COORD_RWINGTIP

It is important that these are placed reasonably accurately in order for ClearView to correctly orientate the model and detect crashes.



In the above picture the three wheel coordinates are being pointed out. These are in exaggerated positions for clarity. For a tail dragging plane such as this the COORD_FLWHEEL should be placed very close just underneath the left wheel and the COORD_FRWHEEL very close just underneath the right wheel. Similarly the COORD_TWHEEL is placed just under the tail wheel, tail skid or in a similar position.

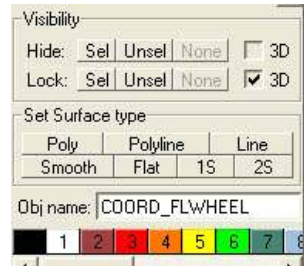
Converting FMS planes for ClearView

For aircraft that have a nose wheel instead of being a tail dragger then COORD_TWHEEL is placed just under the nose wheel with COORD_FLWHEEL and COORD_FRWHEEL under the corresponding main undercarriage wheels.

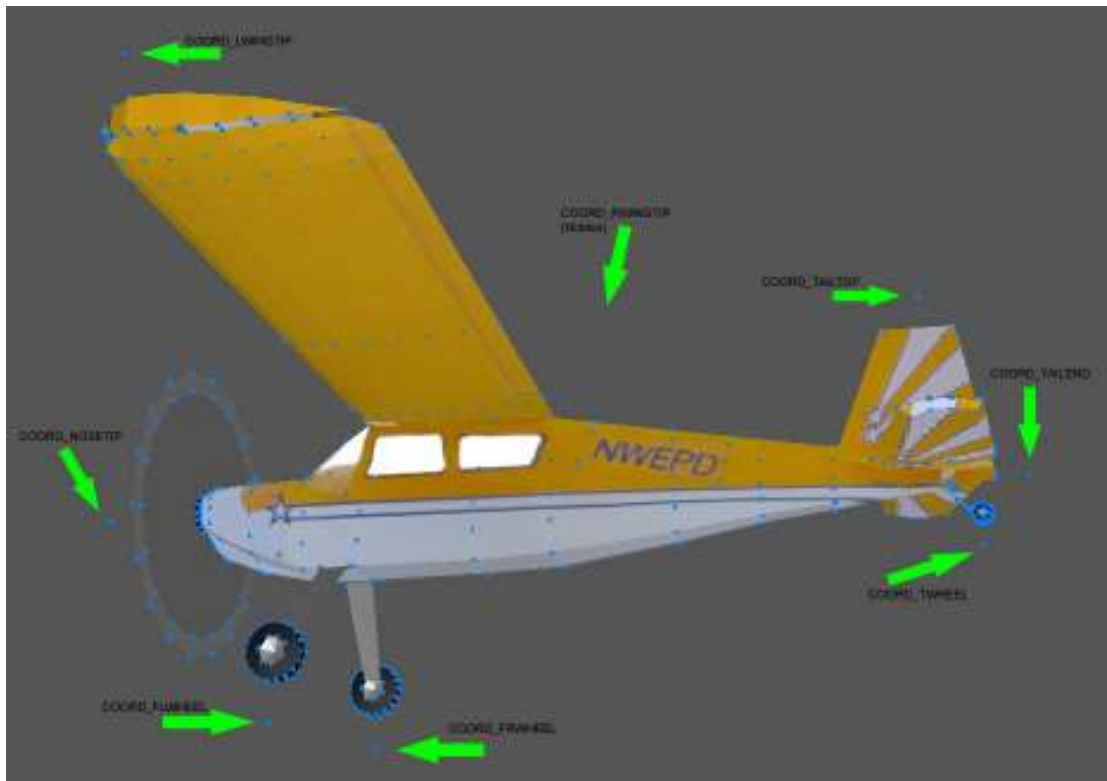
For both types of aircraft there is a parameter that must be set later to tell ClearView if the model is a tail dragger.

Having placed the vertices for the coordinates they must be named exactly as mentioned in AC3D.

Tip: to create a single vertex point needed for the coordinates select object mode and then Line. Click in the position where the coordinate is to be placed and then right click to finish the line. This creates a line with a single coordinate which can then be moved to its required position.



Follow a similar process for the remaining coordinates.



The COORD_NOSETIP would normally go just in front of the spinner or nosecone. COORD_TAILTOP is generally placed near the point that would touch to floor if you turned the plane on its back. COORD_TAILEND is generally behind the furthest back point of the aircraft. Similarly the two wingtip coordinates COORD_LWINGTIP and COORD_RWINGTIP are generally just outside of the corresponding wing tip.

The final element of model manipulation in a basic conversion is to place the model in the correct 3D space. This helps to define the centre of gravity as well as the thrust line. To ensure this is correct move the model in AC3D to align the origin (0,0,0 coordinate) at the expected centre of gravity and thrust line. This can be seen in the following picture.

Converting FMS planes for ClearView



Here you can see the two axes from the front view are on the thrust line and on the wing centre of gravity.

The basic conversion is now almost complete. Whilst ClearView supports moveable propellers and control surfaces you don't need them to fly a model.

As before now is a good time to save the file.

Converting FMS planes for ClearView

Step 5 – Params.txt

Discussion of params.txt values is an advanced topic. However, there are two critical parameters that must be correctly set for ClearView to fly the model successfully.

Params.txt can be found in the SVKSystems/ClearView/models/w-decathlon directory and can be edited with any plain text editor. Alternatively they can be edited by selecting 'Advanced Model Setup' from the settings menu once the aircraft has been selected.

The parameters that must be set are:

- tailDrager
- modelName

If the model is a tail dragger then set the tailDrager value to 1.0 if not set it to 0.0 modelName is set to the AC3D filename of the model. This should be the filename only and not include any directory information. By default ClearView will call this body.ac for an imported aircraft but this can be changed if required.

After you have finished editing params.txt to your satisfaction create a copy called params.txt.bkp and store it in the same directory as the original. This file is used if the user chooses default in the Advanced Model Setup panel. ClearView will overwrite the params.txt with the values stored in the .bkp file.

The model should now be flyable and look as expected in ClearView. At this stage it will not have a moving propeller or control surfaces. If the model does not fly as expected then refer to the discussion on params.txt for help in adjusting flight parameters.

Converting FMS planes for ClearView

Step 6 – Engine Sounds

When ClearView converted the aircraft it uses a default engine sound. For many aircraft with a glow engine this is probably satisfactory. If you have a model that uses a four-stroke, turbine or electric then you probably want to use a different sound.

The decathlon that we are currently converting is an electric powered model so the internal combustion engine sound is in-appropriate.

ClearView uses .wav format sound files. Luckily FMS also uses the same format. If you look in the directory where the original files were extracted you will see a w-decathlon.wav file. If you play this file you will hear an electric motor sound.

To change the model sound in ClearView copy the w-decathlon.wav file to the w-decathlon/wavdata directory. Change its name to plane.wav replacing the file already there with that name. ClearView will play the correct engine sound when flying the model.

You may however notice that although ClearView plays the correct engine sounds it still displays exhaust smoke. Now since this is an electric model, smoke is not a good idea. This can be changed in the params.txt file.

ClearView understands if a model is electric or internal combustion (needing smoke) by examining the 'planeType' parameter. If the parameter is 0.0 this is an internal combustion model and smoke is displayed. If the parameter is 1.0 the model is electric and no smoke is displayed.

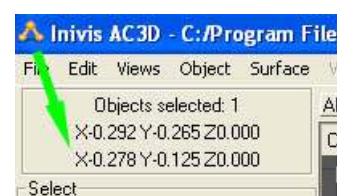
Sometimes the engine sound in the FMS files is not what you desire for your model. Any suitable .wav file can be used as an engine sound though it should be kept relatively short, two to three seconds is generally sufficient. ClearView will just keep repeating the clip indefinitely.

A good source of the correct sound is an actual recording of the engine. Generally a recording of an engine at around half to three-quarter throttle sounds best. If this is not possible then you can always use the sound from another model that has one that you like. Please remember though that you are copying someone else's work and should ask them if they mind and respect their wishes. Look in the model's readme file for the author's information.

Since we are discussing smoke, internal combustion engines (including turbines) generate smoke. ClearView is able to display smoke, the exit point and its direction is controlled by two entries in the params.txt file.

- smokeExostPoint – This is the point at which the smoke starts
- smokeExostVect – This is the direction the smoke initially travels in

Both parameters use an x, y, z reference. Obtaining the setting for the exhaust point is done using AC3D. The easiest way to do this is to create a single vertex as you did for the control points in step 4. You can then easily move this around to get the best start location. The x, y, z position

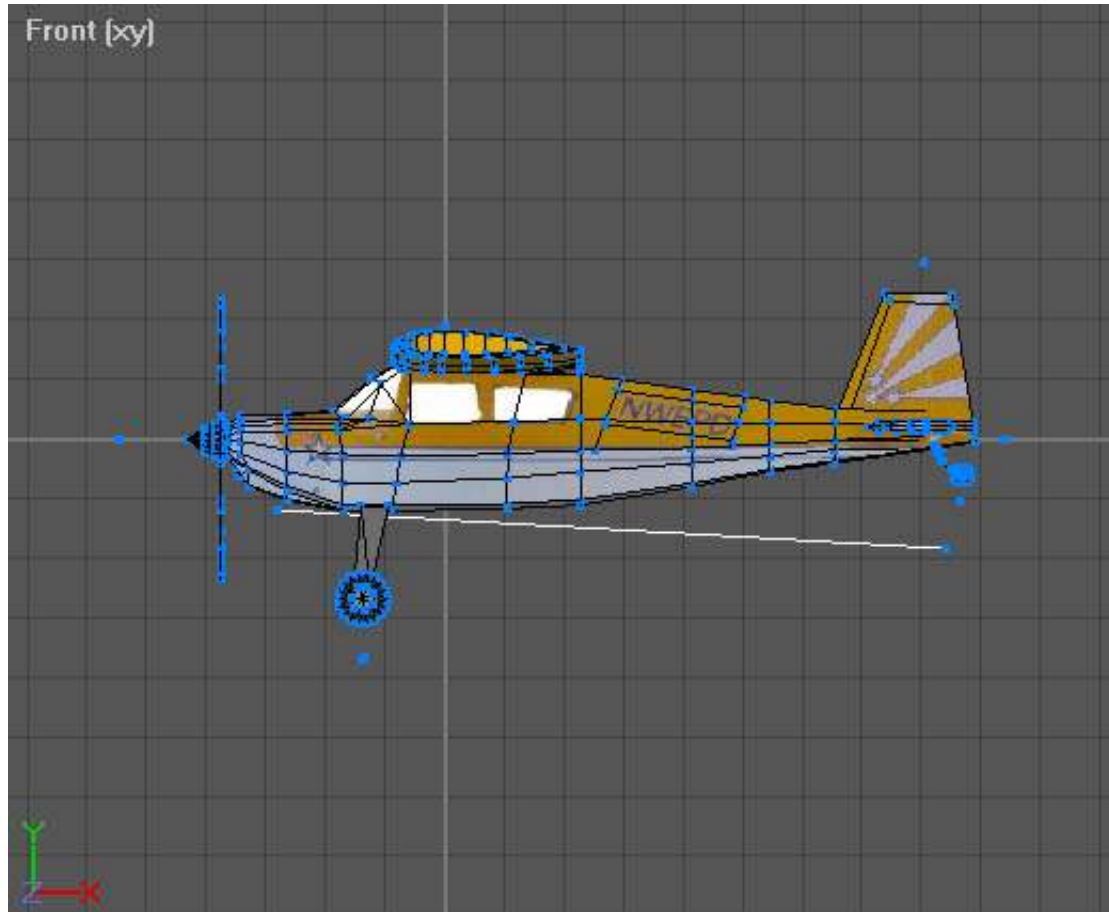


Converting FMS planes for ClearView

can then be read directly from AC3D. The picture shows the coordinates for a point just under the engine cowl. Once you have the coordinate the vertex can be deleted.

Setting the vector is a little harder. A vector determines an end point relative to the start point and is therefore not an absolute figure. There are various ways to determine the vector, with experience of vectors you can set this directly from within ClearView.

Perhaps the easiest way is to use AC3D. Draw a line from the exit point of the smoke in the direction you want it to initially travel. This is shown in the diagram.



This temporary line gives you all the information you require to set both parameters. The coordinates of both ends of the line are:

	X	Y	Z
Initial point	-0.285	-0.118	0.0
Vector point	0.836	-0.182	0.0

To find these points switch to vertex mode and select each of the end points in turn, the coordinates are displayed as discussed earlier.

Converting FMS planes for ClearView

The vector is based upon a simple subtraction. Subtract the initial point from the vector point.

	Vector point		Initial point		Result
X	0.836	-	-0.285	=	1.121
Y	-0.182	-	-0.118	=	-0.064
Z	0.0	-	0.0	=	0.0

This gives you a vector of 1.121, -0.064, 0.0 which then gives values for both parameters of:

- smokeExostPoint -0.285 -0.118 0.0
- smokeExostVect 1.121 -0.118 0.0

The temporary line can then be deleted.

One other point to note is that ClearView will only generate a single smoke stream. If you have a model with two or more engine exhaust outlets then place the initial point at a position further back behind the exhausts. This will simulate where the streams would naturally join together.

Note: If you examine the parameters of the included models and some add-on models you will see that the smokeExostVect has been normalised to have the largest vector as 10. If you wish to do the same you can simply scale the individual values appropriately.

Converting FMS planes for ClearView

Step 7 – Adding control surfaces and props

Preparing

This is perhaps the hardest part of a conversion as it usually involves some complex manipulation of the model in AC3D and the construction of some new objects.

How difficult this is depends upon how the model was constructed and also where the control surfaces are placed. In the case of the decathlon adding ailerons and elevator is relatively simple but adding the rudder requires more complex manipulation in AC3D.

First of all let's take a look at which control surfaces ClearView can animate.

- Elevator - Elevator
- LAeleron - Left Aileron
- RAeleron - Right Aileron
- Ruder - Rudder
- LElevon - Left elevon, generally for flying wings or delta planes
- RElevon - Right elevon, generally for flying wings or delta planes
- Prop - Propeller
- PropDisk - prop disk, swept area of the propeller

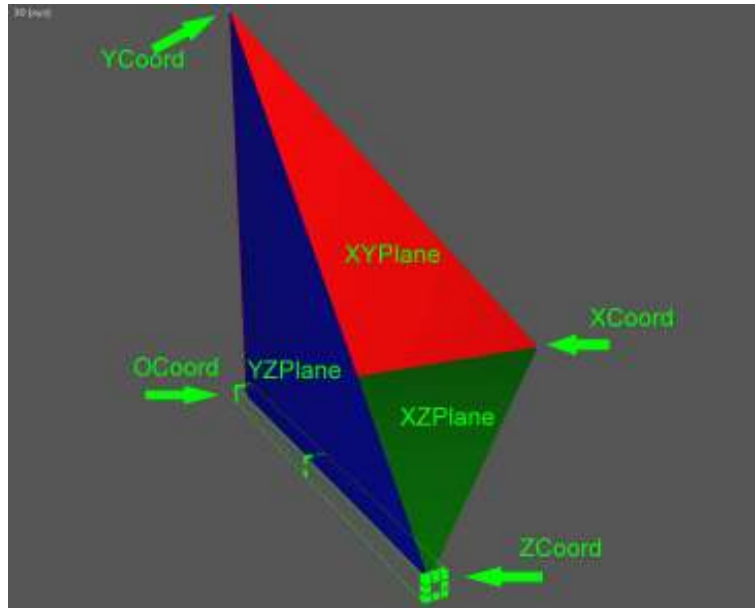
A number of planes have more than one of any type of the control surface so ClearView caters for these as well.

- Elevator1 - Elevator, useful for V type tails.
- LAeleron1 - Left Aileron for biplanes with upper and lower ailerons.
- RAeleron1 - Right Aileron for biplanes with upper and lower ailerons.
- Ruder1 - Rudder for planes with two rudders.
- Prop1 - Propeller, second propeller for planes with two to four propellers.
- PropDisk1 - prop disk, swept area of the second propeller.
- Prop2 - Propeller, third propeller for planes with two to four propellers.
- PropDisk2 - prop disk, swept area of the third propeller.
- Prop3 - Propeller, fourth propeller for planes with two to four propellers.
- PropDisk3 - prop disk, swept area of the fourth propeller.

Converting FMS planes for ClearView

In addition to the control surface ClearView also uses a special group of objects which tell it where the hinge line (point of rotation) is and how the surface moves. This group of objects is referred to as an XYZGroup.

An XYZGroup is illustrated in the picture with the surfaces temporarily coloured for identification. The hinge line is along the vector joining the OCoord to the ZCoord, selected in the picture for clarity. The control surface then moves along the XYPlane.



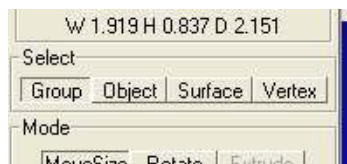
So if you imagine an elevator which is hinged along the OCoord-ZCoord line, the trailing edge would be at the XCoord position. If you now pull full up elevator the trailing edge would move from the XCoord position to the YCoord position. ClearView doesn't actually move the elevator this far but it illustrates the idea.

A control surface is made up of the relevant AC3D object and this special XYZGroup in the correct orientation. A rudder also uses an XYZGroup so you can see that it will need to be rotated to ensure the hinge line and rotational plane are correct.

Now XYZGroups look quite complicated to construct and orientate so let's start by cheating and copy them from an existing model. We will copy them from the 'Edge 540 25 Percent', this model has been chosen because the XYZGroups are in their default orientation.

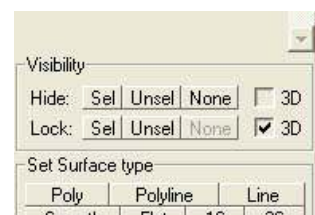
Firstly make a copy of the 'Edge 540 25 Percent' directory because we will be doing major surgery and you don't want to accidentally overwrite the model. (If you do accidentally overwrite something, then simply uninstall ClearView and reinstall it.)

Using the copy that you have just made of the Edge 540, open the Edge540-25.ac file in the data directory. Once AC3D has loaded the model display the Object Hierarchy window (press F8). By default every object will be selected but to be sure press the select all button.

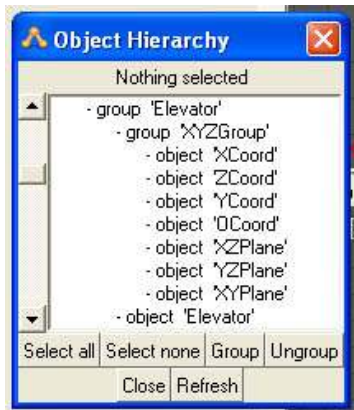


Tip: If you have been doing other things in AC3D before loading this model then you may find that nothing happens when you press select all in the object hierarchy window. If this happens then make sure you are in Group mode.

To make things simpler we will hide everything and then just reveal the objects as we want to work on them. In the Visibility panel at the lower left push the Hide: Sel button. Everything will disappear. Now to display just the item we want to start with.



Converting FMS planes for ClearView



Scroll down the object hierarchy until you see the entry for the Elevator group. Right click on it and select unhide from the pop-up menu. You will see the elevator and its associated XYZGroup appear in the various AC3D windows.

Now switch to object mode and select the object 'Elevator' in the object hierarchy window. It will be highlighted both in that window as well as the other AC3D windows. Now delete the object.

Tip: to delete the object the mouse must be over one of the 'orth' or 3D windows.

You will be left with just an XYZGroup, but it isn't quite usable just yet. Within the object hierarchy window it will now have been moved all the way to the bottom of the list.

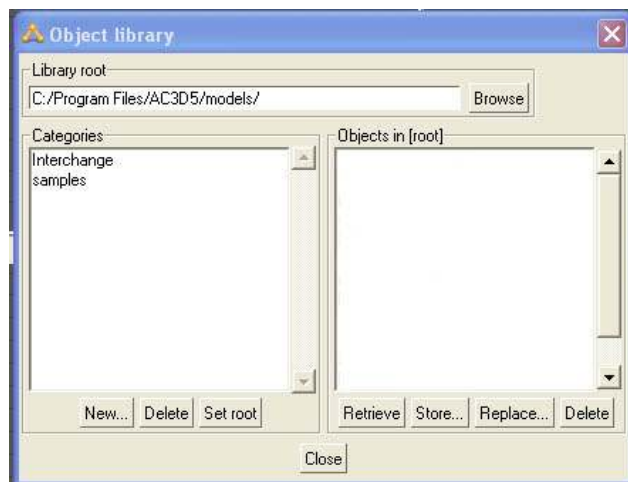
You can see that the group Elevator now only consists of the Group 'XYZGroup'. To extract the XYZGroup we have to ungroup the Elevator. To do this, switch to group mode and select the group 'Elevator'. Then from the edit menu select ungroup. The group 'Elevator' will disappear and you will be left with the group 'XYZGroup' selected.



AC3D has a couple of limitations, you can't just save the XYZGroup as a file by itself as it also includes the rest of the model and you cannot copy and paste an AC3D object between AC3D models (if you have two or more instances of AC3D running). However, AC3D provides a more useful mechanism for achieving the same objectives. It has a library in which you can store components such as this XYZGroup for reuse in many future models.

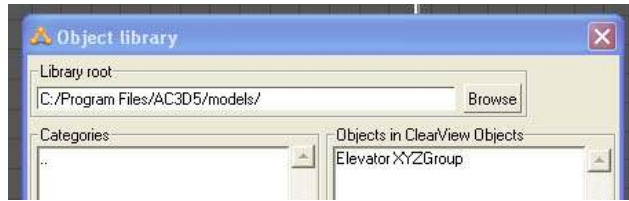
To display the library select 'Object Library' from the File menu.

It is good practice to separate out the different types or usages of objects to make them easier to find in the future. To do this now push the New button and enter a new category name of your choice, say 'ClearView Objects'. This will update the library to show a new category. Like most file windows you can double click on the new category to display the items within it. To start with the category will be empty.



Converting FMS planes for ClearView

Now push the 'Store' button and a new dialog box will be displayed. Enter the name you would like the object to be called in the library. I suggest you use 'Elevator XYZGroup', ensure that 'Store selected' is



checked as we only want the XYZGroup and not the whole model. Then click Store.



Now repeat the same action a further four times to store an object but this time use the following names:

- LAeleron XYZGroup
- RAeleron XYZGroup
- LElevon XYZGroup
- RElevon XYZGroup

Although the XYZGroups are identical this helps avoid confusion when you are first learning to convert planes and also helps as a useful reminder of what things need to be called for ClearView to recognise them.

Now dismiss the object library and then click Hide: Sel to hide the XYZGroup. We now need to repeat the same exercise with the Ruder group, the Prop group and the PropDisk group. Each time adding the final XYZGroup to the library with the appropriate name:

- Ruder XYZGroup
- Prop XYZGroup
- PropDisk XYZGroup

In extracting the Ruder XYZGroup you will need to delete two objects, the object called 'Ruder' and the object called 'TailWheel'. Although we will cover it later this is an example of how a tail wheel and the rudder can move together in ClearView.

To extract the Prop XYZGroup delete the 'Prop' object and to extract the PropDisk XYZGroup delete the 'disk' object.

This will then leave you with eight objects stored in your library.



Converting FMS planes for ClearView

You now have all the basic control groups that you will need to finish converting the Decathlon and any future models.

Tip: you can use the library to copy objects from one model to another, create another category for the purpose to avoid confusion.

Now close AC3D without saving the changes. Although you are only working on a copy of the real model (you did use the copy didn't you?) not saving changes you don't want is good practice just in case.

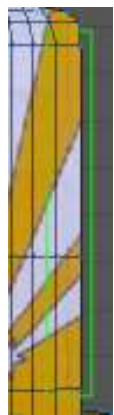
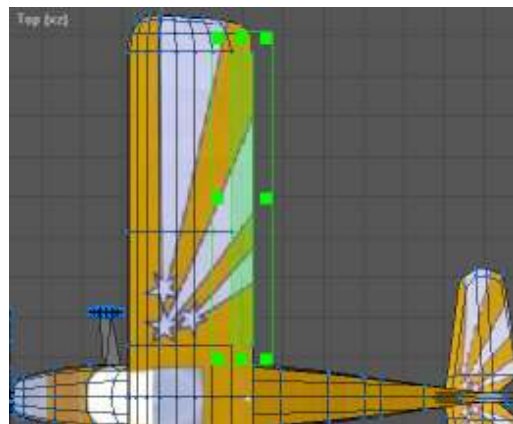
We are now ready to animate the control surfaces of the decathlon. The work you have just done to obtain the XYZGroups only needs to be done once. For future models you already have the XYZGroups so you can skip this stage.

Cutting out the ailerons

Let's start with the ailerons. I am going to make the ailerons strips but not to go the full span of the wing. The portion I'm going to use is shown highlighted.

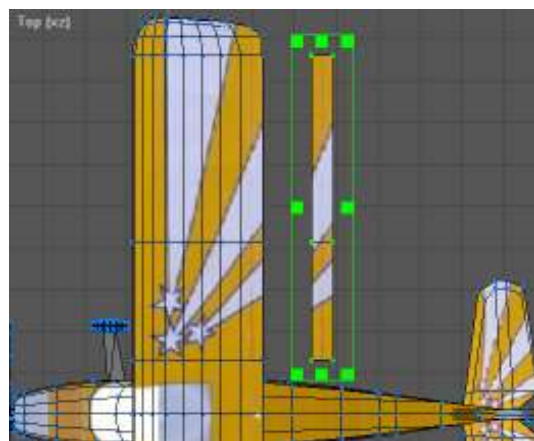
One reason for this choice is that the original author has made it easier because they have placed vertices for the top surface of the wing directly over the bottom surface vertices.

The second reason is that at this stage I don't want to add the complexity of extending the cut line all the way to the wing tip.



To start the cutting process, switch to vertex mode and select the area of the right aileron. It is important that you do the selection using a bounding box (click and drag to surround the vertices) not by pointing at individual vertices. By using the bounding box you simultaneously select both the top and bottom surface vertices.

With the vertices selected perform a copy and paste (edit->copy, edit->paste) you will initially see no difference. You can now drag the selected elements to the right as in this picture.



If the drag distorts the model then it is because a new object was not pasted. To correct this select edit->undo (CTRL-Z), this will then undo the drag. You can then try the copy and paste again.

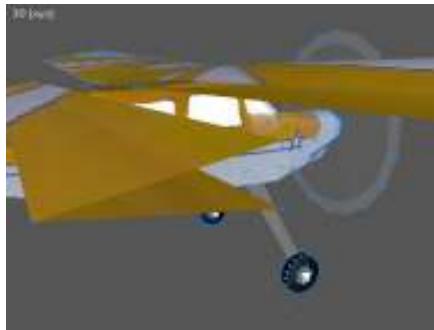
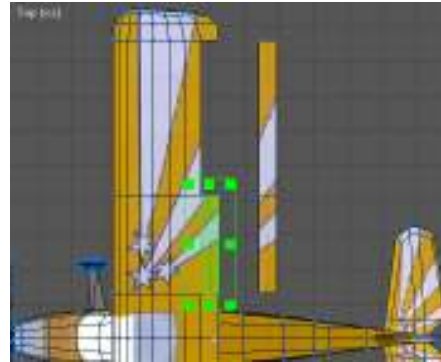
If you now switch to object mode you will be able to rename the object to 'Right Aileron'.

Converting FMS planes for ClearView

Now that we have our right aileron we need to cut out the space on the wing to fit it. The simplest way to do this is to delete surfaces. Switch to surface mode and select one of the surfaces where the aileron will go and then push delete.

As this will only delete the top surface you may not seem much difference, this is because you are now looking at the back of the bottom surface, so select that and delete it also.

Keep repeating this until all the necessary surfaces have been deleted. At this point you could just use things as they are but you may notice some odd visual effects later in ClearView if you do.



If you take a closer look in the 3D window you will notice that the new aileron and the space where it came from seem to be missing some surfaces so you can see inside.

It is a good idea to correct this. If you leave it then ClearView will often give you glimpses of the inside when you manipulate the surface which looks strange.

Let's start by putting the surfaces onto the aileron. To make this easy hide everything except the right aileron. Use the object hierarchy window (F8) to select all then push Hide: Sel. Unhide the right aileron by right clicking in the object hierarchy window and selecting unhide.

You will now begin to create some new surfaces. Generally this is simpler to do using the 3D window as it is easier to see that you have selected the correct vertices.

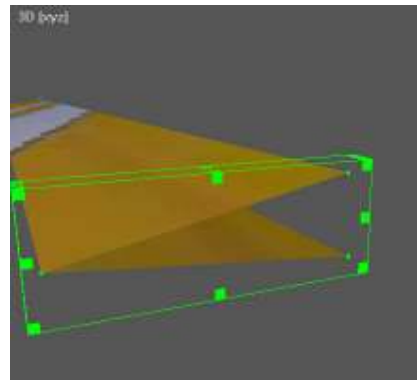
To create a new surface we need to select the three or four vertices that will make up the new surface. Whilst it is possible to create a surface using more ClearView will not display it and it will have to be triangulated.

Start with the right hand edge of the aileron and individually select each of the three vertices.

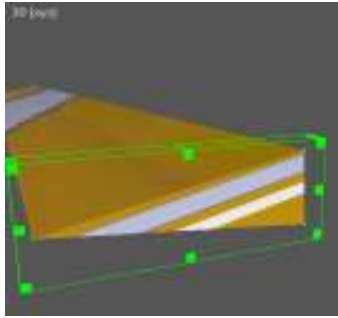
Tip: if you can't see the vertices you can display them by selecting vertices from the 3D menu.

Tip: hold shift when selecting additional vertices.

Although it doesn't matter which vertex you chose first it is important that you select the rest going anti-clockwise. This ensures that the 'normal' is in the correct direction. Don't worry if you do it wrong you can always flip the 'normal' as discussed earlier.



Converting FMS planes for ClearView



You now create the surface by selecting 'Create Ordered Surface' from the Vertex menu (Ctrl-Shift-S).

You will notice that a texture has been applied and although this is probably not the look required we can change it later. For the moment leave it and create the remaining three surfaces of the aileron.

The long front of the aileron is made up of two surfaces. Don't be tempted to just create a single surface using the four corner vertices as this creates an object that has a hole in it, somewhat analogous to not gluing it in the middle. At times you will be able to see the very thin gap so it is something to be avoided.



A good idea at this point is to turn on the display of 'normals' and check that they are as they should be, correct any that are wrong.

Let's now correct the texture that has been applied to the new surfaces. In this instance we just want the surfaces to be completely yellow and the same colour as the rest of the wing. Although you can do all the new surfaces in one go let's do them one at a time for experience.

Start with the right hand edge of the aileron. Switch to surface mode and select the surface. Now open the texture editor (Tools->Texture Coordinate Editor or F10). At first you may not see anything in the window so zoom out to see the whole image. You can also select surface and then push Select All to see a bounding box with the surface highlighted.



Now at this point you could go to the trouble of correctly laying out the triangle but since we just want the whole surface to be yellow if we make the triangle extremely small we can then place it anywhere on the yellow areas.



To do this change to Select by: Vertex and the push Snap: Together. You then have what looks like a single point which can be dragged to sit over an area of yellow.

If you take a look at the 3D window you would now see that it has become full yellow.

Now repeat this exercise for the remaining surfaces of the aileron.

With the right aileron completed hide it and unhide the wing. You can then do much the same actions to fill in the missing surfaces in the aileron gap.

Converting FMS planes for ClearView

With the right aileron completed you can repeat the actions on the left wing to create the left aileron. Finally with both ailerons complete you can unhide everything and, switching to object mode, move the ailerons back into their normal position.

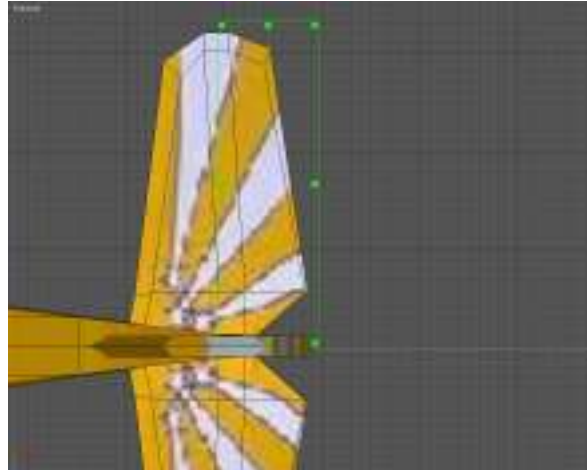
Cutting out the elevator

Now for the elevator, exactly the same techniques are used to create the elevator. If you take a close look at the tail plane you will see a natural line for the elevator.

Although it is not straight it is close enough that the difference will not usually be noticeable and using it as it is saves a lot of work.

Again the original author has used vertices that lie above one another so we can use vertex mode and a bounding box to select the relevant vertices.

Tip: if you can't position the bounding box to encompass all the vertices in one go you can use two or more bounding boxes, remember to keep the shift key pressed when you create the next bounding box and AC3D will add them all into the selection for you.



As you did for the ailerons create a right and left elevator. Each elevator needs four surfaces to be created and three surfaces in the gap it came from. Incidentally if you take a look at the textures when you have created the surfaces they match quite well to the pattern on the top and bottom so you may wish to leave them like that rather than making them all yellow. Which ever you prefer is correct. At this point save the model because we are next going to add new capabilities to AC3D.

Cutting out the rudder

Although the rudder is a simple straight design there are no suitable vertices in the correct position so it is necessary to use a different technique to obtain it.

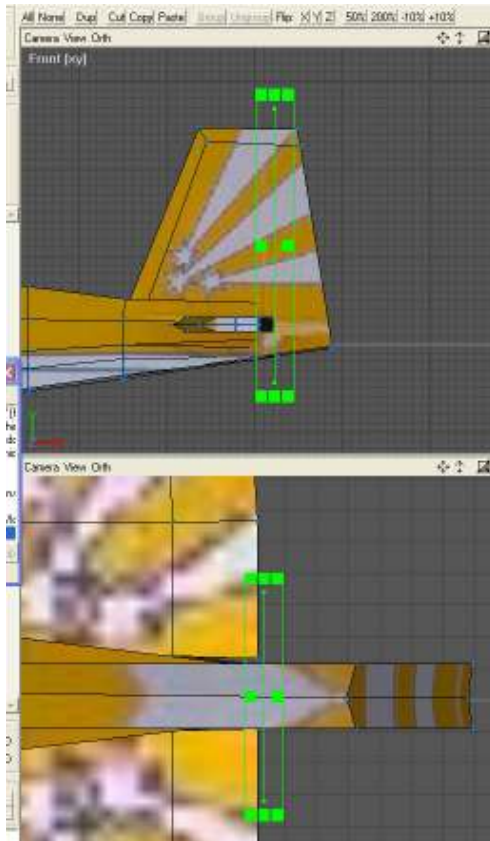
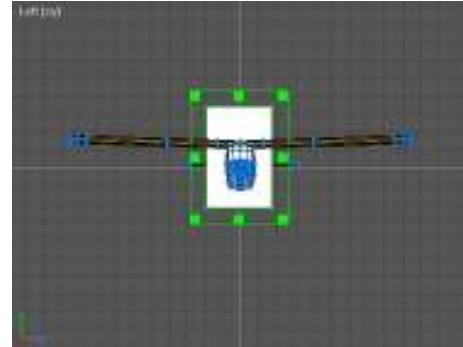
There is an AC3D add-on tool which will make detaching the rudder easier. It can be found on Dennis Hawthorne's website. You can find details by going to the online forum from the help menu or directly from this URL:

<http://supercoldmilk.com/ac3dplug/index.html>

Look through the page for the 'Knife' add-on (not the Ortho Knife). Download and install it as described in the instructions, there is also a short description of how to use it. I will describe how to use it in relation to obtaining our rudder.

Converting FMS planes for ClearView

Using object mode, create a rectangle in the left (or right) view. By default it will be created along the z-axis. Move this back towards the position of the rudder and reduce it's width in the 'left' window. It is good practice with a knife to make it just big enough to slice what you want. A knife which is too large sometimes slices other things that you don't want sliced. It can also help to hide the left and right elevators to make it easier to see.



Drag the knife until it is just behind where the elevators join the tailplane. In the picture the elevators have been hidden to make this positioning easier. The knife has also been reduced in size so we can see that it will only cut where we want to cut.

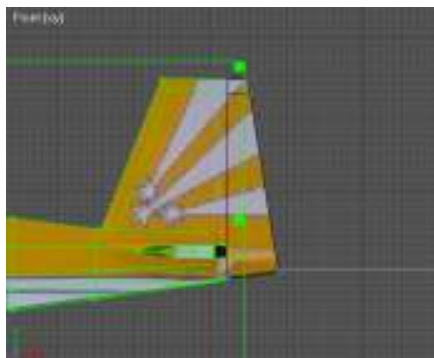
Now display the knife control panel by selecting 'Knife' from the Tools menu.



If the knife is not currently selected ensure that you are in object mode and select it. Now push the 'Set knife' button in the Knife control panel and its name will appear in the panel.

Ensure that all three tick boxes are checked as we want to slice the surfaces, cut away the rudder as a separate object and since we don't need the knife again it can also be deleted.

Now click on the rudder to select the object we wish to slice and push the 'Slice' button on the knife control panel. You can then dismiss the knife control panel.



You can now see that the rudder has been cut away and the rest of the model remains selected. The rudder can now be selected and dragged away. This is also a good time to name the object as a rudder.

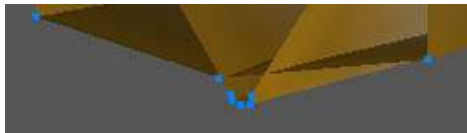
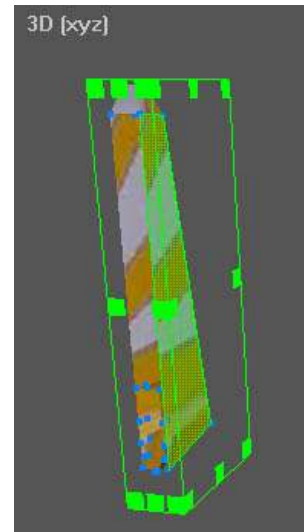
Converting FMS planes for ClearView

As with the elevators and ailerons, this has left the rudder and fin with some missing surfaces. Add these surfaces exactly as you did for the ailerons and elevator. Be aware though that in this case there are more vertices and the surfaces are a little more complicated.

When you look at either the rudder or the fin you will see a lot of vertices. Deciding which ones to use to create a surface can be difficult so one tip is to see which vertices are being used on the outside surface.

Switch to surface mode and select one of the outside surfaces. You can see that it goes virtually from top to bottom and which vertices it uses. All the rest inside are left over from chopping the back end off of the plane. If you feel confident you can delete them. If something goes wrong you can use Edit->Undo to correct the mistake.

The rudder only requires three or four surfaces to complete it. The difference depends upon whether you use a four vertex surface or only use triangles (three vertices).



When it is time to add the surfaces to the body you may be left wondering which one you should use. With care you can see that of the group in the middle it is the upper left. But this is not really satisfactory as it leaves a small hole. A much better solution here is to merge all the centre vertices into a single vertex.

Select all the centre vertices. You can do this in the 3D window by holding Ctrl whilst creating a bounding box around them.



Then select 'Snap Together' from the Vertex menu (Ctrl-T). Not only does this tidy up the back of the fuselage but it also makes it easier to add in the missing surfaces which you can now do as before. Correct any textures if they don't look as you would like them to. Finally, unhide everything and drag the rudder back to its correct position.

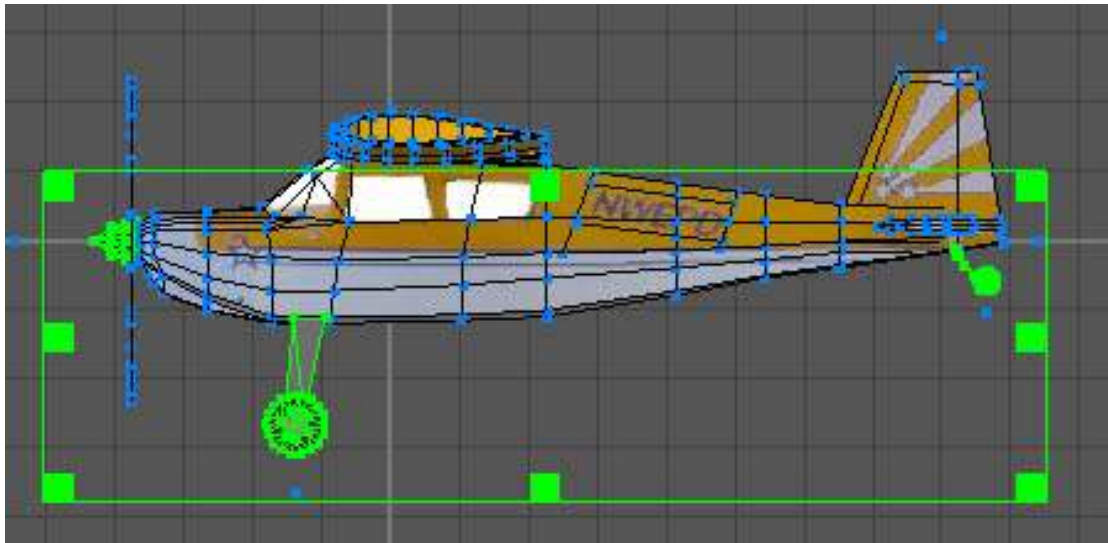
Cutting out the tail wheel

We are almost ready to animate the control surfaces but since this is a tail-dragger and it has a nice tail wheel, it would be nice if it moved in response to steering inputs (rudder) as well as the rudder. To do this we have to extract it as a separate object in the same way that rudder, elevator and ailerons have been. There are various ways to do this and I will illustrate a different technique that can sometimes be used to good effect.

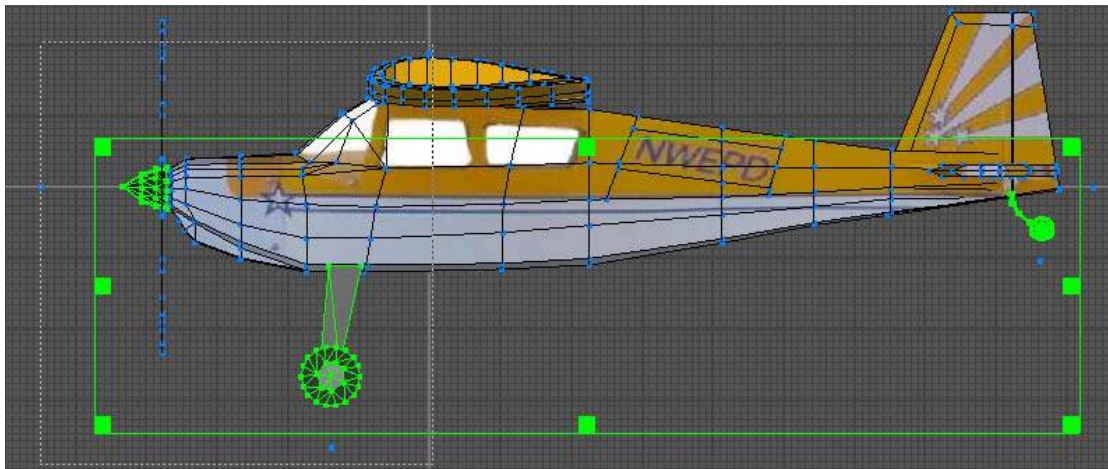
If you switch to object mode and click on the tail wheel tyre, you will see that it forms part of the same object as the main undercarriage tyres and the spinner. If you click on the tail wheel undercarriage leg you will see that it is part of the same object with the main undercarriage legs.

Converting FMS planes for ClearView

Selecting just the tail wheel and undercarriage would seem difficult so we will use a little trick. Use a bounding box to select the undercarriage leg and tail wheel. This will also select the front undercarriage, wheels and spinner.



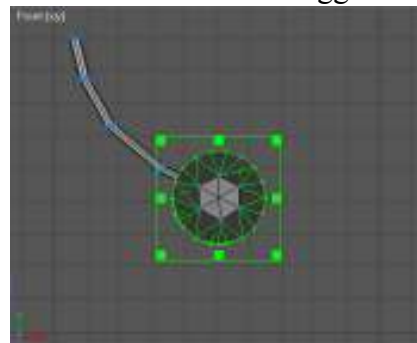
Now switch to vertex mode and deselect all the vertices associated with the spinner and front undercarriage and wheels. Deselecting is done by using the right mouse button whilst holding the shift key. It is most easily accomplished by using a bounding box.



This will then reduce the selection to just the tail wheel and its undercarriage leg. A simple copy and paste will duplicate the assembly as before. It can then be dragged to one side. Repeat the selection process to once again select the tail wheel and leg before deleting it.

You now have two objects, a tail wheel and an undercarriage leg. It is better if these are both combined into a single object.

In object mode select both the wheel and the leg. Then select 'Merge' from the object menu. Since we have just merged two objects into one it is a good time to check for redundant vertices and invalid surfaces.



Converting FMS planes for ClearView

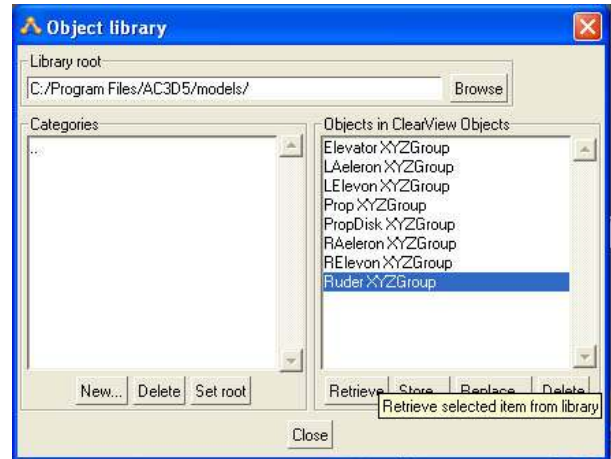
From the Object menu chose 'Optimize Vertices'. This should remove 24 vertices that are no longer required. Choosing 'Optimize Surfaces' should result in no surfaces being removed.

Finally rename the object as the 'Tail Wheel' and drag it back into position as part of the rudder. Save the file before we start animating in case anything goes wrong.

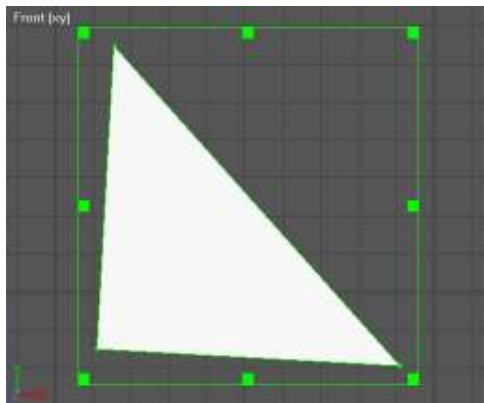
Animate the rudder

We will start by animating the rudder because it is the simplest in this model to start with. Ensure that nothing is selected and then display the object library, (File->Object Library). Change categories to display the ClearView Objects that you created earlier.

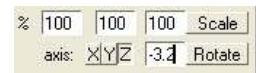
Select the Ruder XYZGroup and click Retrieve. The 'orth' windows will now change to display the XYZGroup.



At this point you will notice that the group isn't quite vertical in the front window. As this is the hinge line it will need to be corrected.



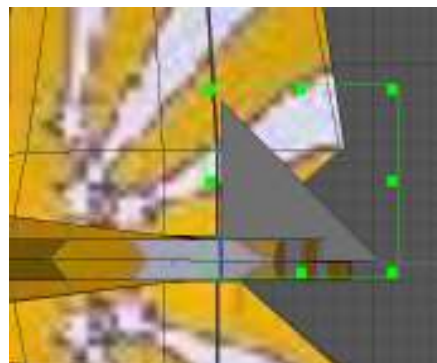
To correct this tilt ensure that AC3D is in 'Group' mode and then rotate the group -3.2° around the Z-axis. This straightens up the group.



It is a good idea to store this modified version back into your library for next time so assuming you still have the library displayed click 'Replace'. It will prompt you that the object already exists and confirm that you really want to replace it. Click 'Yes' to

replace it. You can now dismiss the library.

Ensure that you are in group mode and then drag the XYZGroup until it is in line with the hinge line of the rudder. In this case I am choosing to hinge the rudder along its centre line.



Converting FMS planes for ClearView



Now using the object hierarchy window, also select the object 'Tail Wheel' and the object 'Rudder'. Then select 'Group' either from the edit menu or by pressing the 'Group' button on the object hierarchy window. This will create a single group called 'group'. This should now be renamed 'Ruder'.

The rudder control now consists of the rudder itself, the tail wheel and an XYZGroup that tells ClearView how to move things in response to a control. If you save the model now you would be able to load it into ClearView and observe the rudder, and tail wheel, responding to

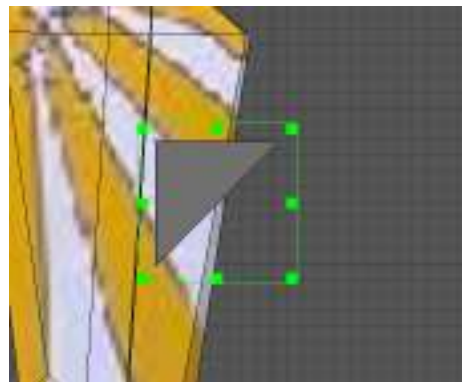
rudder control.

Now we can basically replicate these same actions for the elevator and the ailerons. However, there is a slight twist to the elevator. Often the elevator hinge line is at right angles to the centre line of the body. In the case of this model the elevator hinge lines are swept forward.

With right angled hinge lines it would be possible to use a single Elevator XYZGroup to indicate the hinge line and then just associated both elevator halves with it in a similar way to how we associated the rudder and tail wheel. If we were to do this for this model then one of the elevator halves would move in a very strange way when viewed with ClearView. This is one case where we use two Elevator XYZGroups, one for each elevator half.

Starting with the left elevator drag the XYZGroup until it is over the left elevator. You may need to raise it above the elevator to see the whole of the triangle.

As you can see the group is not quite aligned with the elevator hinge line. To do this the group should be rotated 3.5° around the Y-axis.



It can then be dragged to the hinge line and moved down through the elevator until it rests in your desired hinge line, top, middle or bottom hinged.

Once in the correct position, again using the object hierarchy window, also select the left elevator and then group them together. Call the group 'Elevator'. Repeat the same actions for the right elevator only this time it will need to be rotated -3.5° around the Y-axis. Group it with the right elevator and call the resultant group 'Elevator1'. Save the model as you now have working elevators and rudder.

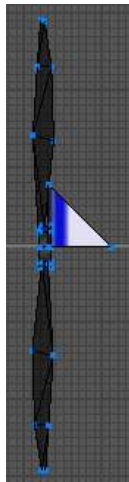
Converting FMS planes for ClearView

To complete the ailerons you complete basically the same actions. This time the LAeron XYZGroup needs to be rotated -2° around the X-axis and when combined with the left aileron called 'LAeron'. Perhaps not surprisingly the RAeron XYZGroup is rotated 2° around the X-axis and when combined with the right aileron should be called 'RAeron'.

Again at this stage save the model as you now have the Decathlon with fully moveable flying surfaces. All that remains is to animate the prop.

Creating an animated propeller

If you are confident in your use of AC3D you can go ahead and create your own propeller from scratch. You have an indication of the prop diameter from the FMS propdisk that is still attached to the front of the Decathlon. If you are not that confident then let's borrow one from the Edge 540 we used earlier.



Open the copy of the Edge 540 we used earlier. If you don't want to close the Decathlon you can start another instance of AC3D to do this. As before, using the object hierarchy, hide everything and then unhide the prop group. Select the group and then un-group it. This will leave you with a group 'XYZGroup' and an object 'Prop'. This time select the XYZGroup and delete it leaving you with just the prop.

Ensure the prop is selected then display the object library. It is a good idea at this point to create another category in which to place objects like this when you are copying them between models. In this case I called the category 'Copy'. Store the prop to this new category ensuring you chose to 'Store Selected' as we only want the prop.

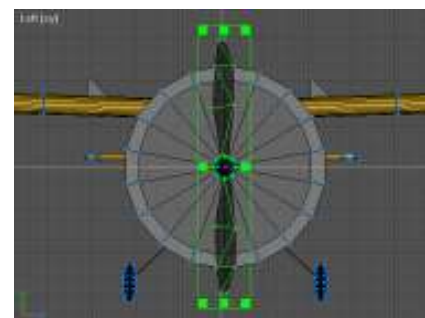


Whilst we have the Edge 540 open hide the prop and then do the same again for the prop disk deleting its associated XYZGroup. Before storing it we need to remove the texture associated with it. So whilst it is still selected chose Texture->Set No Texture from the object menu. Now store it into your copy category.

You can now close the Edge 540 without saving changes and return to the Decathlon.

Using the object library, retrieve the prop from the copy category. Drag it until it sits in front or just inside the spinner. You may notice at this point that the existing spinner and prop disk are off-set. We can correct this later if you prefer it centred.

You will also notice that the prop is too big and must be resized. Although this looks like a matter of simply dragging the bounding box to reduce the prop diameter you should resist the temptation. Doing it in that way will result in a propeller that looks wrong, it is too fat for its length.



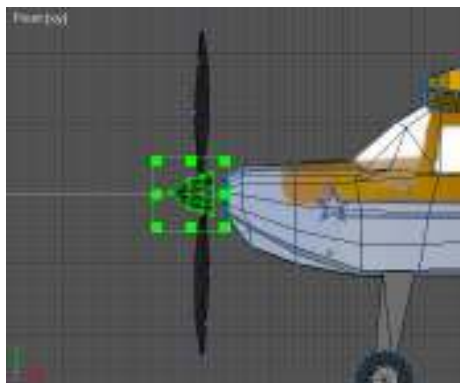
Converting FMS planes for ClearView

You should reduce the size of a 3D object by scaling it. A scale of 80% is about right in this case. You can now temporarily hide the prop.



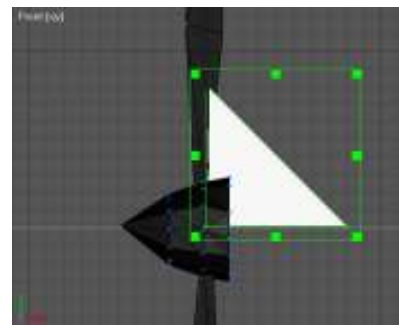
Let's now get rid of the existing prop disk as we no longer require it. As with the control surfaces the easiest way to do this is to switch to surface mode and delete the individual surfaces until they have all gone.

Before we insert the new prop disk we need to separate the spinner. You may recall that it is part of the same object as the front wheels. Refer back to the section on extracting the tail wheel and follow the same principles to create an independent spinner.



Unhide the prop and position it with the spinner until you are happy with the two together. Using the object library retrieve the Prop XYZGroup you created earlier and move it to the propeller.

It is important that the XYZGroup is on the centre line of the propeller otherwise it will wobble. Generally it is also placed so that it

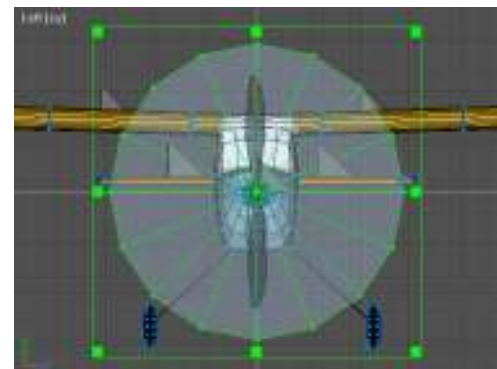
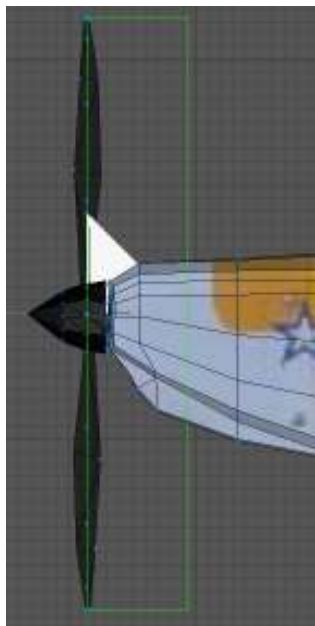


lines up with the radial centre line of the blades but this is less important.

As previously switch to group mode and create a group called 'Prop' consisting of the prop, the spinner and the XYZGroup. Whilst in group mode you can now move the Prop group to your preferred position on the model.

You can now use the library to retrieve the prop disk. As with the prop it is larger than we need. A simple scale will reduce it to the size we need. If

you remember that this was the prop disk for the propeller you will have guessed it needs to be scaled by 80%.



Retrieve the PropDisk XYZGroup from the library and move it to the propdisk. Ensure that the OCoord is at the centre of the prop disk and then create a group containing the both and called 'PropDisk'.

Finally move the PropDisk group to coincide with the centre line of the propeller blades.

Converting FMS planes for ClearView

Some final tidying up

If you take a look at the original readme you will see that it contains some useful information. The readme provides the overall length, wingspan and weight of the Decathlon.

One unit in AC3D roughly corresponds to 1 metre. If you check the two vertices that mark the two wingtips you can determine the model wingspan. By subtracting one Z-axis coordinate from the other you will obtain a wingspan as roughly 2 metres. From the readme it should be 730mm.

I will leave the rescaling of the model as an exercise to be picked up in a more advanced tutorial. If you try this on an internal combustion model then it would be necessary to determine a new smoke exhaust point from AC3D and enter it into the params.txt file. There is no need to change the exhaust vector as this is still valid.

Now all you need to do is to save the model, close AC3D, load ClearView and enjoy your model.

The conversion is now complete.

With experience of trying to fly the model, or if you are converting a model you have actually flown you may wish to modify any of the 72+ parameters that affect the flight performance. Some of these will be covered in a more advanced guide.

For the moment if you try flying your model you will notice it flies as fast as a jet and has the aerobatic performance of an Edge 540.

Converting FMS planes for ClearView

Step 8 – Packaging

Before packing the model for distribution it is necessary to complete two further files. As mentioned earlier, copy the 'params.txt' to 'params.txt.bkp'. This is a backup file so that a user can default the parameters back to their original values if they have been changed too far.

The second step is to complete a readme file. What you put into this file is entirely up to you but it should contain at least you contact details so that anyone wishing to make use of your work will be able to ask permission.

As an example the readme.txt for my distribution of the Decathlon is contained below.

Watt Age Decathlon EP ARF

This Model was converted from an FMS model by Mark Glennon using AC3D and is freely distributed. Please feel free to modify as you wish using AC3D. If you modify and make it fly more realistically, feel free to contact me via email, as I would love to see improved versions created for ClearView RC Simulator, particularly an accurate flight model. I can then update the distribution to make the flight model more realistic for everyone.

To install, simply extract the file and put the folder under the models in the ClearView RC Simulator. The next time you run ClearView, it will appear under the Airplane selections.

Mark Glennon

Mark@zaragon.plus.com

The original FMS model was created by Gary Gunnerson and the original readme is below.

Model: Watt-Age Decathlon EP ARF (yellow)

Email: garyg51384@aol.com

Web-Site: http://gunnerson.homestead.com/files/fms_models.htm

Created: June 2005

Notes: The Watt-Age Decathlon EP ARF airplane is an almost ready-to-fly scale looking airplane. The Decathlon comes with an all wood built-up wing construction and real iron-on covering. It comes with all the ancillary hardware horns, rods, gear, etc. but you'll need to add a complete flight pack, motor, batts, servos, ESC and receiver.

<http://www.globalhobby.com/public/gallery/128414.asp>

Wing Span: 28.75 in / 730mm

Overall Length: 24.25 in / 616mm

Flying Weight: 16.2 oz / 460g

Motor Size: 370 power with gearbox required

Radio: Proportional 4-channel required A/R/E/T

Servos: 3 micro servos required

Battery: 8-cell 9.6V 800mAh Ni-MH required

If there are any Watt-Age Decathlon owners out there, please try the model with FMS and let me know how it flies compared to the real thing.

-- Gary

Converting FMS planes for ClearView

To distribute the file create a zip file containing the contents of the w-decathlon directory including the sub-directories. If you are using a compression application to create the zip file ensure that it retains the directory structure.

If you are using Windows XP, it has the ability to create Zip files. The w-decathlon directory can then just be dragged and dropped into the zip.

Finally ensure that the zip file has a meaningful name to make it easier to distinguish when you have lots stored on the PC.

The END.