

Modelling

Supported Modelling tools

Currently (30th January 2017) the Aerofly FS 2 Software Development Kit ([SDK](#)) supports 3D Studio Max 2010 or higher, Maxon Cinema 4D version 17 and AutoCAD (AC3D) 3D version 8.

Robin DR-400 - Example Aircraft

In the [SDK](#) we have included an entire aircraft project, the Robin DR-400, including its 3D model and textures. By following the few steps of the [aircraft tutorial](#) the necessary development programs will be installed and at the end of the tutorial the DR-400 will be installed as an additional aircraft in the simulator.

Please note that the 3D model of the Robin DR-400 provided is copyrighted and may not be used for any other purpose than demonstrating how to create aircraft for the Aerofly FS 2.

Preparing a Custom 3D-Model

After installing the [SDK](#) we will now go through the steps of converting a custom made 3D model from 3Ds Max into the Aerofly FS 2 Flight Simulator.

To achieve an error free export some preparation is needed to be done first. For the DR400 example aircraft all these steps have already been completed so that this plane can be used as a reference.

Single 3D Model

Aerofly FS 2 works with just one 3D for an entire aircraft; there is no need to provide an external and internal model separately. The Aerofly FS 2 engine is optimized to draw only those objects that are in the view of the camera.

Generic Modeling Units

The units of the aircraft model should be set to 'Generic Units'.

Alignment and coordinate system

The aircraft should be aligned to the x-axis as the forward flight direction, the positive y-axis towards the left and the positive z-axis pointing upwards. The origin of the model coordinate system should be placed close to the center of gravity.

Animated objects should have their axis and rotation pivots set to a plausible location. Hinged objects should have their local origin in their hinge location. The pivots and axes set in 3D Studio Max will be exported to the log files which will be used later on to animate the objects in the graphics section of the .tmd file.

Animated objects should be positioned in the following neutral positions before export:

- control surfaces neutral, flaps up
- gear down and compressed as if the aircraft would stay on the ground
- gear doors closed to assure perfect alignment when the gear is retracted in flight
- cabin doors and windows closed
- knobs and switches that are not planned to be animated should be in an in-flight position
- switches and knobs that will be animated are best exported in their off state
- instrument needles point to zero or neutral depending on the instrument
- magnetic compass points towards 090 degrees (East). This is needed because the x-axis of the global coordinate system of the Aerofly FS 2 points towards the East as well.

Individual objects

All animated objects like the control surfaces (rudder, aileron, elevator) and trim rudders, switches and knobs, gear parts and so on have to be individual objects in the 3D model so that they can be animated independently from each other. The Aerofly FS 2 rendering engine is optimized to merge objects in the same GeometryList when the aircraft is loaded, so there is no real benefit combining objects. Experience has shown that development time can be reduced if all switches and knobs are available as individual objects right away and the model doesn't need to be changed every time a developer decides to animate a certain switch.

Display screens have to be separate objects so you can set the display brightness independently of the background lighting later.

Object naming conventions

3D objects will be referenced by their name for display and animations, so it is essential that each object has a unique name. Objects should be named logically, and the name must not contain spaces or special characters (it is recommended to use A-Z, a-z, 0-9 only). While the object names have no meaning to the simulation, it is easier for everyone working on the project to use standardized names. We use the 'UpperCamelCase' convention, i.e. we capitalize the first letter of each word to get short but readable object names.

Advisable names are e.g.:

Fuselage, LeftWing, LeftAileron, LeftFlapInner, LeftSlat1, LeftSpoiler1, LeftSpoiler1LinkUpper, LeftStabilizer, LeftElevator, VerticalStabilizer, Rudder, LeftGearUpper, LeftGearLower, LeftFrontWheel, LeftPropeller, LeftBlade1, LeftBlade2, VirtualCockpitStatic, SwitchBatteryMaster, MCDUPushButtonA, CanopyFrame, CanopyGlass, COM1FrequencyKnobSmall, NAV1FrequencySwap, LeftEngineFan, LeftEngineExhaust, LeftEngineRevererFlap0, PrimaryFlightDisplayPilot, NavigationDisplayCopilot,...

Materials

Materials are defined by the set of textures they use and by tags or hints that describe the intended usage. A simple material just needs a diffuse texture, but Aerofly FS 2 also supports ambient, specular, bump/normal, reflection, (self-)luminance and illumination textures to define the material parameters at each texel. Use the 3D software's standard material and the following texture slots:

- Diffuse: diffuse color. An optional grayscale '_alpha' texture can be provided to add transparency information (black is fully transparent).
- Ambient: ambient color. This is a separate grayscale RGB map.
- Bump: bump map if texture name ends with '_bump', normal map if name ends with '_normal'. The bump maps of a model will be converted to normal maps by the aircraft converter.
- Specular: specular highlight color and strength. Color and strength are given by RGB values of the texture, specular highlight width is controlled by the alpha channel
- Reflection: strength of sharp reflection. Red color channel controls the strength of the environment reflection. Do not use the green and blue channels as they are reserved for other purposes.
- Self-Illumination: color luminance map.
- Filter Color: illumination. RGB channels encode the illumination strength/lightmap for the interaction of the diffuse color with three independent light sources. (This slot is used in 3D Studio Max only. In most cases, we don't assign these textures explicitly but use the suffix '_illumination' and let the converter add the texture to the material.

We use the suffixes '_color', '_ambient', '_bump', ... to distinguish the different textures that make up one material. These suffixes are not required but allow the aircraft converter to automatically add textures to a material if they have the same base name, and the 3D modeling software doesn't support a particular texture slot.

Textures

All textures must be square power-of-two images with a pixel depth of 24-bit RGB .bmp, 24/32-bit .png or .tif files. Bump maps can be 16-bit grayscale textures. There is no support for .psd files in the converter, the different layers for diffuse, ambient, .. have to be exported to the intermediate folder first. You can keep a .psd file assigned to a texture slot in the 3D software if it has the name as the bitmap file except for the suffix.

It is recommended to create all textures at a resolution of 4096×4096 or even 8192×8192 pixels, and use the appropriate converter setting to reduce the actual size used in the simulator. The smallest size recommended is 32×32 pixels, the maximum supported texture resolution by the engine is currently 4096×4096px. The total compressed texture amount should be in the range of 150 MB to ensure compatibility with all platforms. The final texture resolution can be set in the 'model.tmc' file.

Refresh Texture assignments

If necessary, refresh the texture assignments in the 3D modeling software to use the exported textures in the intermediate folder. Again, you can keep a .psd file assigned to a texture slot in the 3D software if it has the name as the bitmap file except for the suffix. The converter ignores the texture suffix, so you can change from .bmp to .tif without changing or exporting the 3D model again.

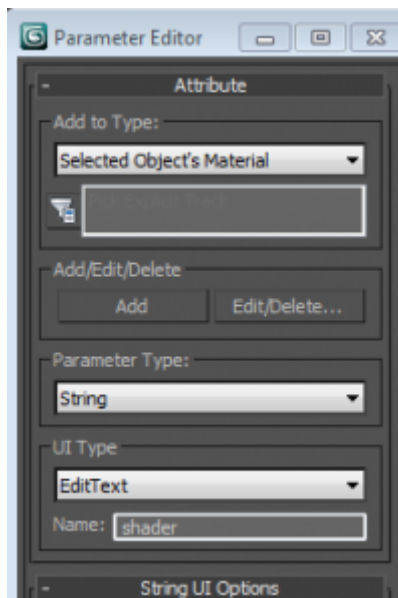
Material Tags or Hints

Each material can define tags or shader hints to support the selection of the best shader when the aircraft is loaded. Shader hints can be combined to describe the material properties as precise as possible. Currently, the following shader hints are supported:

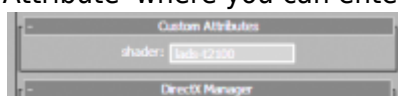
| | |
|------------|--|
| exterior | everything that is not in the cockpit/cabin, outer faces of windows and canopy |
| interior | everything that is in the cockpit and cabin, inside of windows and canopy |
| default | |
| standard | |
| glass | |
| canopy | |
| window | |
| instrument | |
| darken | |

The hint can be added to a material in 3D Studio Max as follows:

1. Select an object that uses the selected material
2. Open 'Animations' → 'Parameter Editor'
3. Select 'Add to Type' → 'Selected Object's Material'
4. Select 'Parameter Type' → 'String' and 'UI Type' → 'EditText'
5. Enter 'shader' in the 'Name' textbox
6. Click the 'Add' button



- The material now has a 'Custom Attribute' where you can enter the desired tags.



Exporting from 3D Studio Max

If you haven't done this yet, install the export plugin as described above. Load the aircraft in 3D Studio Max and select 'Export' → 'Export' and select the aircraft's intermediate folder. Select 'IPACS TGI Exporter (*.TGI)' and save to 'xxxx.tgi'. Select the output option 'Aircraft model' in the exporter menu and set the scale factor to

- 10 if the one generic unit in 3D Studio Max corresponds to 1 mm
- 100 if the one generic unit in 3D Studio Max corresponds to 1 cm
- 10000 if the one generic unit in 3D Studio Max corresponds to 1 m



After all parameters have been entered a click on the button 'Export' will launch the export process. The export summary will notify the user about any errors in the export process and will summarize the total polygon and vertex count and other statistics.



Converting a 3D-Model to the Engine-Format

When you have collected all required files for an aircraft in the intermediate folder, run the aircraft converter: right click on the 'xxxx' intermediate folder and choose Aerofly FS 2 Aircraft Converter from the context menu. This opens up the converter window and the aircraft's internal name xxxx should be displayed. Start the conversion. If run for the first time, the converter will convert and compress all textures and render the previews. This might take some time depending on your hardware.

The next time you convert the aircraft, the converter will only process only those textures that have been changed. You can always force the conversion of a texture or preview by deleting the generated file.

If you want to change the output folder, you can set this in the 'config.tmc' file located in your 'Documents/Aerofly FS 2 Aircraft Converter' folder. Change the entry 'DesktopFolder' to match your folder.

The same folder contains the converter's log file 'tm_aircraft_converter.log', which sometimes is

helpful for troubleshooting, as well as the 'report.txt' in the aircraft's intermediate folder.

Aircraft 3D Modeling Guidelines

We use the following 3d modeling guidelines for the aircraft in Aerofly FS 2 and provide them for reference. These requirements are not mandatory and you can get an aircraft into Aerofly FS 2 without adhering to each requirement (none of our models adheres to all). However, experience has shown that these best practices make it easier for all people involved in the project and create models that run smoothly on all platforms. If you ever plan to release your aircraft for different platforms, it is strongly recommended to fulfill as many requirements as possible.

Summary

An aircraft model for Aerofly FS 2 includes the exterior and interior in one model. The overall polygon budget is 400k triangles and 200k vertices. The total compressed texture amount is in the range of 150 MB; this allows up to 2 GB of uncompressed textures at a resolution of 4069×4096. A unique, non-shared, non-degenerate texture mapping is required for ambient occlusion, illumination and texture baking. The mesh has a clean topology without T-junctions and long triangles. The dimensions are correct and the model is set up using generic units, where one generic unit corresponds to some metric unit.

The following guidelines should clarify what we consider being a clean model and should help to integrate a model into Aerofly FS 2. None of our aircraft models adheres to all items entirely, but this is what to aim for to have the finest aircraft in Aerofly FS 2.

Mesh

- Single model that contains both exterior and interior (there is no separate virtual cockpit)
- Overall polygon budget: 400k triangles (not polygons!) / 200k vertices for everything.
- +X axis is forward, +Y axis points to the left, +Z is up
- Center of the coordinate system is approximately at center of gravity (don't have the wheels at $Z = 0$)
- Aircraft has to be modeled in the clean configuration: flaps up, control surfaces neutral, gear doors closed, wing unloaded (no wing flex in the 3d model, this will be done by the sim). Landing gear needs special treatment: this is modeled in extended and loaded position.
- Switches, knobs, pointers, ... : model in a position that corresponds to their position in normal flight configuration, i.e. engine switches on, temperature and oil pressure needles in green arc, systems switches auto / on, transponder on and encoding, fuel and packs on, ...
- Plan ahead and make a polygon budget for different sections (fuselage/gear/wings/cockpit/...) don't start at a small detail, then work up at the same detail level and end up with too many triangles
- No single object larger than 50k triangles / 50k vertices.
- Get the dimensions right from the very beginning, this is very hard to fix later. Don't start with some small detail and work up from there.
- Use generic units in 3ds max. One generic unit corresponds to either 1mm, 1cm, 10cm or 1m real world size (all other units in max are language dependent and will change if opened on another computer with different settings)

- Depending on the dimensions of the real aircraft, use one of the scalings given above that fits your workflow best (consider near/far clipping in max, camera movement, displayed coordinates, enough precision for small details, ...)
- Check pilot's seat and controls position and dimensions using a pilot's model at an early stage.
- Object naming conventions:
 - Name starts with an uppercase letter, followed by a-z, A-Z, 0-9 only.
 - No spaces, no underscores, no minus/hyphen '-', no special characters whatsoever.
 - Capitalize the first letter of a word (camelback): LeftWingLight2, FrontGearDamper, ...
 - Use expressive names, don't use abbreviations in object names like LtLtKnob, use LeftLightKnob
 - Each object has to have a unique name.
- Anything that will be animated separately has to be a separate object. If in doubt, use more objects and give them reasonable names.
- Clean meshes:
 - No t-junctions
 - No long triangles
- There are no hard edges in reality: use 45 deg bevels of 0.5 - 1.0 mm in the cockpit where necessary. Apply smoothing groups 31 or 32 to the bevel and use hard edges, Aerofly will create smooth normals and leave the normals of the adjacent faces intact (or use your favorite plugin to achieve this, sometimes this is preferable for rendering of ambient occlusion maps)
- Support wing flexing: use divisions/cuts running from the leading edge to the trailing edge across the wing and flaps/ailerons.
- Skinning is supported by applying vertex weights. Use a bone system with Bone 0 and Bone 1 to apply the weights.
- Use animations for control surfaces, flaps, landing gear, levers, knobs, ... to check the model and help setting up the aircraft in Aerofly. Keyframe 0 is the frame that will be exported, make sure object positions and orientations are as defined above.
- Keep future versions in mind: e.g. different versions with different fuselage lengths.

Textures

- Only use power of two square textures: 32×32, ... , 256×256, ... , 4096×4096
- Supported texture file types: BMP (24bit), TIF / PNG (24/32/16/48bit)
- Total texture amount:
 - compressed textures 35 MB for mobile (mainly 1024×1024)
 - 150 MB for desktop (mainly 2048×2048) this corresponds to about 2 GB of total textures for uncompressed 4096×4096 bmps / tiffs (about 40 4096×4096 textures or six main materials)
 - some materials don't need large maps, usually you will end up with 20 - 30 materials
- Texture default size should be 4096×4096 for main maps. However, keep in mind that text has to be readable at the 1024×1024 for mobile devices (cf. UV mapping guidelines below)
- RGB color values 0 .. 255:
 - use values in the range 8 .. 16 for black surfaces, values smaller than 8 only for holes
 - 232 corresponds to a pure white, values above 232 (in any channel) will add a fluorescent/neon look to the material.
- Use compression friendly color values for large uniform areas.
- Compression noise can be suppressed by adding some noise (use cautiously).
- Use noise, dirt, scratches at those locations where it makes sense, don't apply these everywhere
- Don't use parts of photos as textures.
- Don't bake illumination/highlights into textures.

- Keep textures mipmap friendly: no bright objects on a dark background and vice versa, leave room for a little border, adjust background to object color.
- Bumpmaps are pieces of art of their own, don't create them simply from an image or color layer.
- Same holds for specular (color+alpha): these are not just black and white versions of the color layer.
- All maps for one material go into a single psd, use groups for specular, ambient, color, ...
- Example for grouping in psd file:
 - UVW Templates
 - Checkerboard
 - Illumination
 - Light/Luminance
 - Reflection
 - Specular
 - Ambient
 - Bump
 - Alpha
 - Dirt
 - Markings
 - Integral lighted
 - Unlit
 - Details
 - Paint
 - Airlines
 - Repaints Background (differently colored patches to support mipmapping)
- Keep rendered layers (ambient/illumination) in psd, don't edit these layers (re-render if necessary)
- Use 16 bit bump maps where necessary to avoid terracing.
- If you prefer to work with bmp files, you can use `_alpha` to add an alpha channel to a rgb bmp: `xxx_color.bmp` and `xxx_color_alpha.bmp` will be combined into a four-channel texture

UV Mapping

- Plan ahead: use your texture budget reasonably.
- Unique mapping is required for ambient occlusion and illumination.
- UV sharing:
 - only for identical geometries exactly at the same UV location (clone objects after UV mapping)
 - only if these objects never need different ambient occlusion or illumination
- Uniform mapping:
 - minimize stretching/distortion
 - use the same resolution/pixel density in world space for objects close to each other use checkerboard textures to verify
- Pay attention to the pilot's view: resolution for instrument panel and objects in pilot's view: 4-6 pix/mm for 4096×4096 textures.
- Exterior UV mapping: fuselage should be mapped contiguously on one texture for aircraft up to the size of an A320, cut into two parts for larger aircraft. This defines the resolution for other exterior parts like wings, stabilizers, engines, ...
- Keep the uv mapping repaint friendly. Use a small texture for pieces in the cockpit that can change with a repaint (e.g. registration number).
- Use grouping of similar objects for better mipmapping.

- Group objects reasonably: don't put small objects onto another texture just because there's some space left.
- Leave a border around objects for better mipmapping.
- Preferably map all parts visible from the main forward view onto the same texture, we can boost the size of this important texture if everything in the main view gets sharper.
- Clean mapping:
 - no degenerate UV faces
 - minimize cuts
- Use a separate texture for displays (display_luminance) and things that will be animated.
- Use tiling for some materials/layers to save texture memory without sacrificing resolution (the underlying mapping must still be unique for ambient occlusion).
- Don't squeeze too much into one texture, leave room for mipmapping, baking and future changes/additions.

Decals

- Use decals for small details only. Don't use decals for large logos.
- Place 0.2 - 1.0 mm in front of mesh.
- Underlying mesh still must be closed and uniquely mapped (so we can render the decal into the base map if necessary)
- A smaller and lower resolution version of the decal can be included in the base texture (to avoid popping if decal is culled at some distance or with z-fighting)
- Don't cut decal into the underlying mesh.

Materials

- The standard material supports:

| Function | 3ds max slot | Texture file suffix |
|------------------------|-------------------|---------------------|
| diffuse / color | Diffuse Color | _color |
| specular color + width | Specular Color | _specular |
| reflection/gloss | Reflection | _reflection |
| ambient occlusion | Ambient Color | _ambient |
| bumpmap | Bump | _bump |
| (self-)luminance | Self-Illumination | _luminance |
| illumination | Filter Color | _illumination |

- Materials can be tagged to select the best shader:
 - exterior
 - interior
 - glass
 - standard
 - default
 - darken
 - canopy
 - instrument
 - window
- Multi-materials are deprecated: no need to use them.
- Every material should correspond to exactly one material ball in max's material editor, all

materials must be visible in editor.

- Use bump mapping (normal maps are supported, bump mapping is more flexible and better suited for future gimmicks). Object space normals are not supported.

File Names

- File and folder names: allowed characters are lowercase letters a-z, digits 0-9 and underscore only.
- Never use capital letters, spaces and special characters in file names.
- Don't include the aircraft's name in texture names (don't use 'f15_xxxx_color.bmp').
- Current max file is named like 'f15_2015-06-18_56.max'.

From:

<https://www.aerofly.com/dokuwiki/> - **Aerofly FS Wiki**

Permanent link:

<https://www.aerofly.com/dokuwiki/doku.php/sdk:aircraft:modelling?rev=1486372372>

Last update: **2017/02/06 10:12**

