

ShowboxN documentation

Introduction

ShowboxN is a program that is used to assemble models, textures, and different animations into a binary model for in-game use in Ground Control II (GC2).

Setup

Working directory. All files used in GC2 are referenced using relative paths. First start the ShowboxN and goto Tools -> Options and make sure that the working directory points to the GC2 main folder. Click Save to store the settings.

Unpack necessary files from the GC2 .sdf files. If you want to use content from GC2, you must first unpack those files from the .sdf files so they are located directly on the hard drive. For instance, if you want to use the file `gc2data.sdf:units\fl_cc\cc_marine_new\cc_marine_aim_test.mrb`, you must unpack it to:
`(workdir)\units\units\fl_cc\cc_marine_new\cc_marine_aim_test.mrb`.

Controls

Use the left mouse button to rotate view.

Use Control key + left mouse button to zoom in/out.

Use Shift key + left mouse button to pan left/right/up/down.

Function keys toggle states.

File formats

.MRB

The (binary) file format exported to and used by the game engine.

.MRE

The (text) file format that stores the changes made to a model, similar to a project or session file in other programs.

.TGA, .DDS

Image file formats supported by the ShowboxN and GC2. Use DDS in most cases since this is a compressed format, which can also store mip map levels.

A photoshop plugin for saving DDS file is included in the SDK. Start with using TGAs in the beginning to get you comfortable modding, and then you can switch to DDSs later.

.LWS

Lightwave scenes. This is what you will import to get animations, lights, etc.

.LWO

Lightwave objects. These are referenced from the Lightwave scenes. If you only want a simple object without animation or lights, you can import a LWO file directly.

.MS3D

Milkshape3D model. If you don't have access to Lightwave, you can import models created with a simpler (and cheaper) program called Milkshape3D (www.milkshape3d.com). Milkshape3D supports animations and you can also add lights (in a special way mentioned later).

.PE

Particle emitter files. These files are scripts describing how a particle emitter works.

Tutorial 1: Create a simple model for GC2

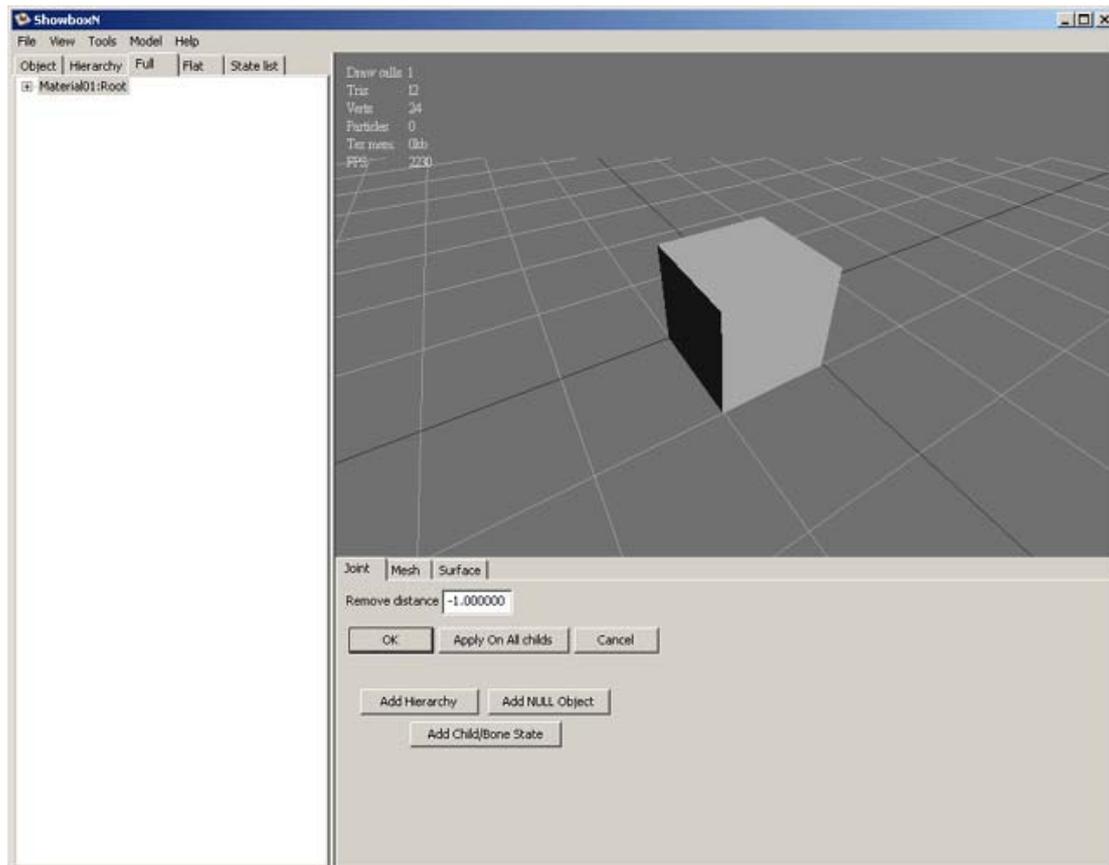
File -> Open...

This will start a new project (MRE). *You will loose the changes to any previous project.*

Choose the source model to import (lws,lwo,ms3d).

Example: **(workdir)mymod\crate_normal_01.lwo**

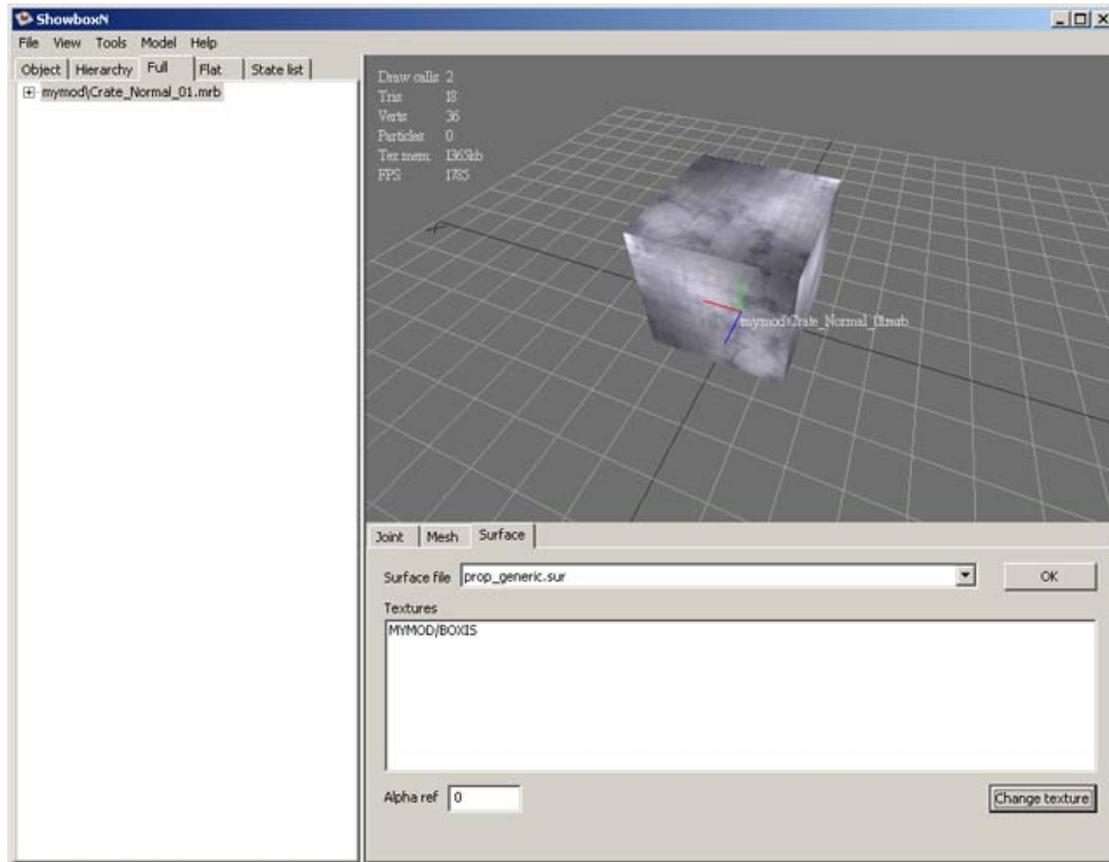
Choose a state (stand) from the list.



Select the tab in the left dialog called Full.

In the bottom right dialog three different tabs appear.

Select the tab called Surface. Choose the surface called prop_generic.sur in the drop down surface selection box. Mark a texture to change and select a texture file.



Now you have the (simplest possible) model that can be rendered in GC2.

Export the model to the binary format by clicking File -> MRB Export ...

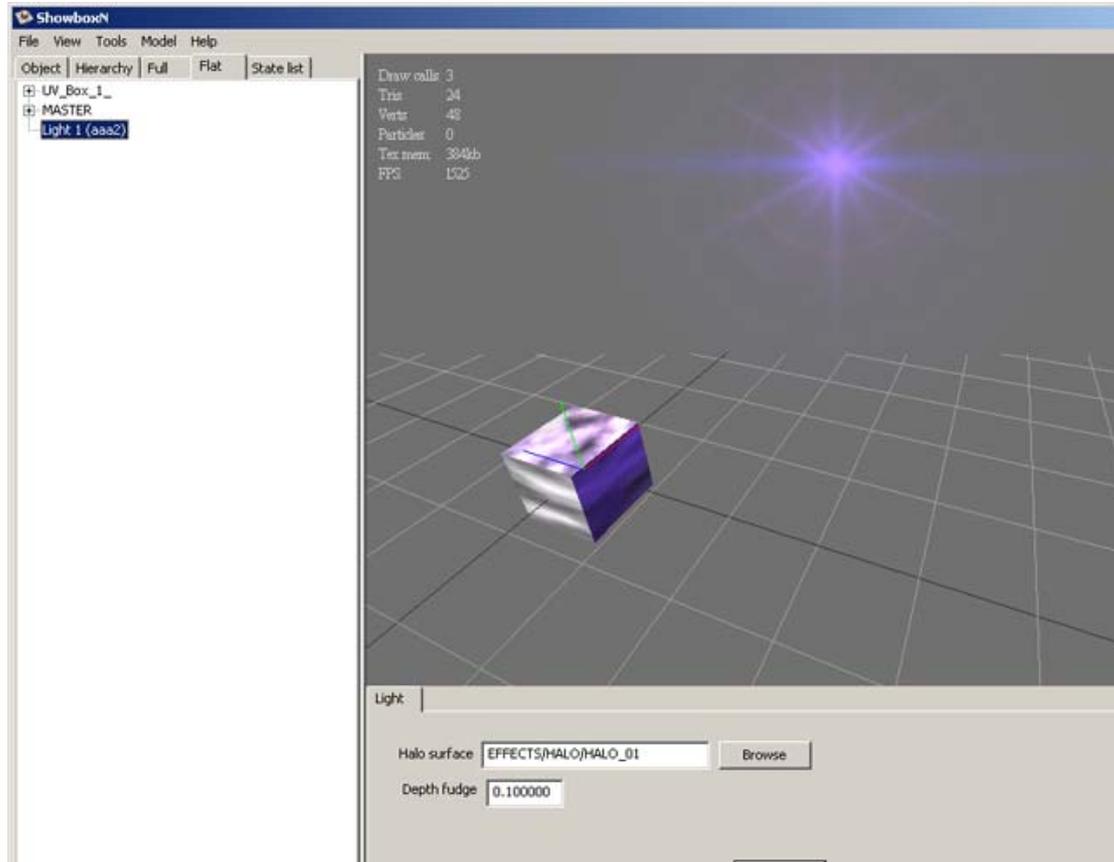
Choose the path (workdir)mymod\mycrate.

With the MRB Export command, both an MRB (binary file for the game engine) and MRE (your models project file) will be saved. *You can at any time save the project file only by clicking File -> Save or File -> Save As...*

Next time you wish to work on this model, open the MRE file!

Tutorial 2: Create a more advanced model

This time we wish to create a little more advanced model which will have animations and a light source. Since animations and lights are defined in Lightwave scene files (.lws), this time we will import an .lws file.



The scene contains a rotating box, and a light source that shines on it.

Light properties.

We can change the halo texture on the light, or remove the halo (by emptying the field and pressing OK). We can also set the halo's depth fudge, which is the offset towards the camera. Depth fudge is used to stop halos from intersecting with nearby geometry.

Lightwave note: Light type *Point light* becomes a light source in the model, and light type *Spotlight* becomes a halo only. These are the only valid light types when importing from Lightwave scenes.

Animations.

When it comes to animation, there isn't much to do about it in the ShowboxN. It should work as it does in Lightwave.

States

One of the most important uses of the ShowboxN is to define states for the model.

Take a door for instance, it has two states: Open and Closed.

Another example is a marine soldier, it may have three states: Stand, Walk, and Fire.

When a state is selected (from the engine code or from a script), that state is set for those bones that have that state. Not all bones must have all the states. Example: A marine's legs have two states: Walk and Stand, while its upper body has three states: Walk, Stand, and Fire.

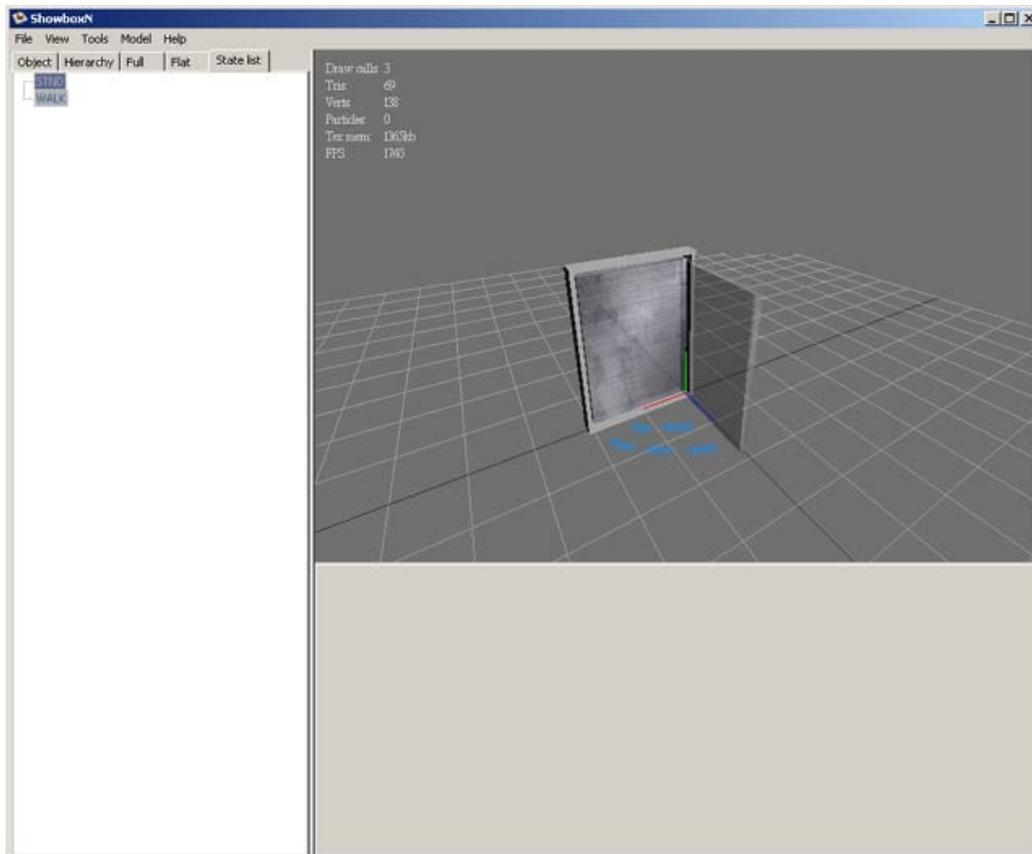
Example: Door

Start a new project by clicking File -> Open...

Choose (workdir)mymod\door_closed.ms3d
Choose state Stand.

Then select the Object tab and choose Add Complete State.
Choose (workdir)mymod\door_open.ms3d
Choose state Walk.

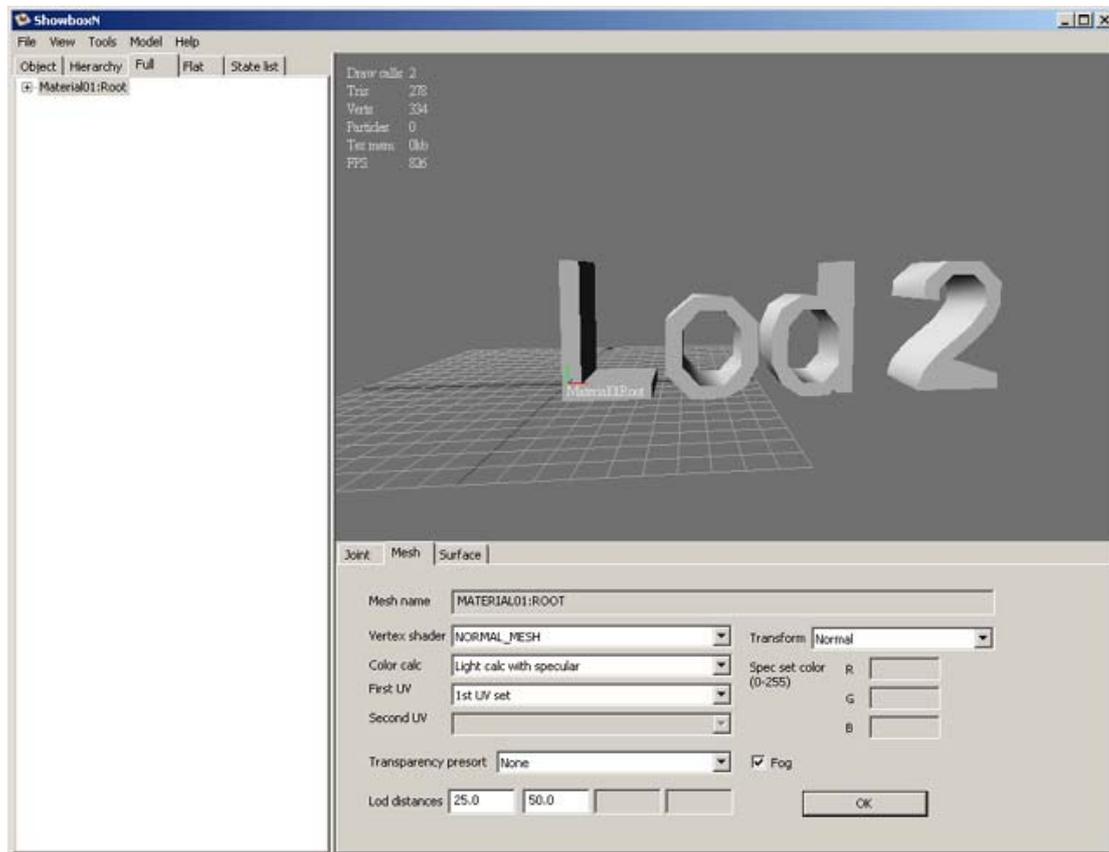
Select the State list tab and click the two states. The door will now interpolate between the two states.



Level-of-detail (LOD) Levels

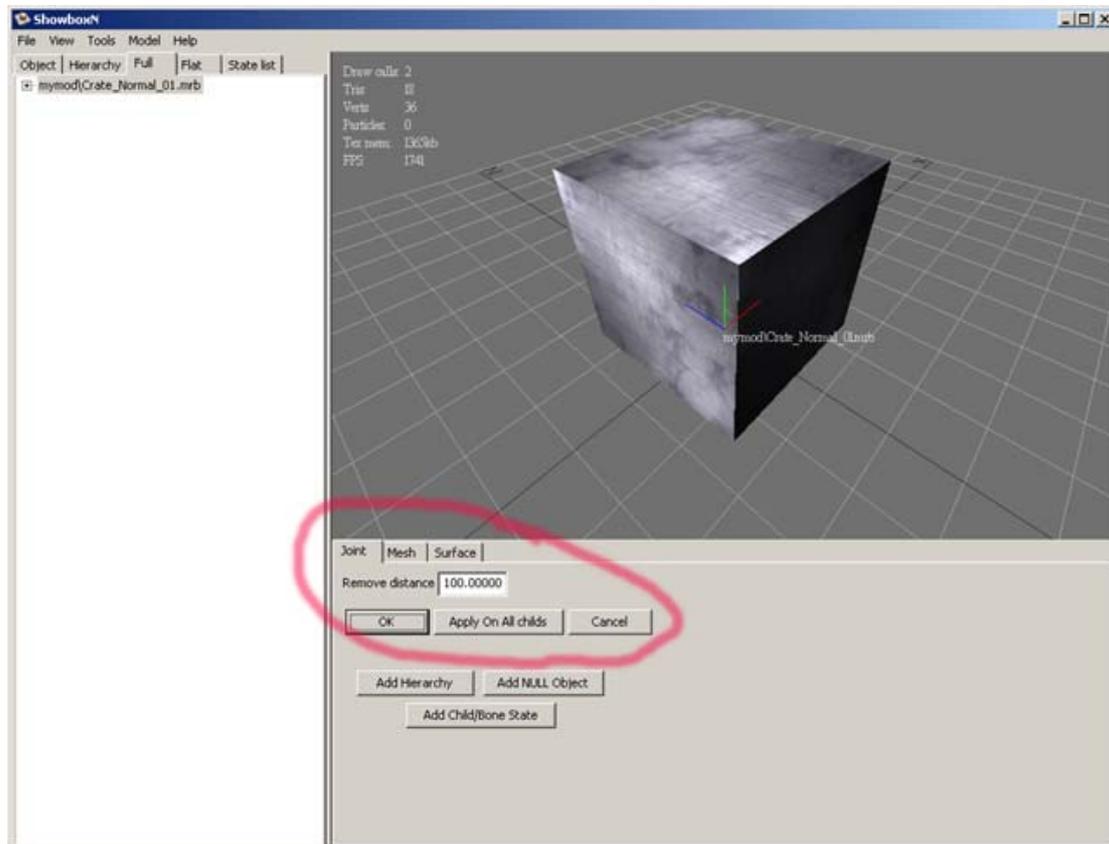
When you load models (.lwo or .ms3d) lower lod levels are automatically included if the models have the `_LodX` suffix (where X is a number from 2-5) in the filename. If lower LODs are found, you can set their change distances in the Mesh dialog.

Example: Open lod.mre to see an example of this.



Remove distance

In addition to separate LOD models, you can also set a distance to fade out and remove a bone/joint.



This joint's remove distance is set to 100.00, which means that the joint will fade out or pop out (depending on surface file) when it is 100 units away from the camera.

Particle Emitters

Particle emitters are scripts that define how a particle system behaves.

Particle emitters are added to states, and they are automatically turned on when a state is enabled, and off when a state is disabled.

To add a particle emitter, do the following:

Open the file door.mre.

Select the Flat tab in the left dialog.

Expand the first joint and select the state WALK.

In the right bottom dialog, click on Add PE. (*The Particle Emitter dialog appears*).

Click on Emitter file Browse... and select (workdir)mymod\me_blacksmoke_01.pe

Click Add.

Now you have added a particle emitter to the state WALK.

Go to the State list tab and toggle between the two states to see how the particle emitter is turned on and off.

Milkshape3D specifics

Lights.

Since the Milkshape3D format doesn't include light sources, we've made a little work-around for this.

To add a light to a Milkshape3D model, do the following:

Add a joint with the naming format:

LightXX(R,G,B,RANGE)

...where XX is a number (00 – 10)

...and RGA are in 0-255 range (i.e. white is 255,255,255).

Animation.

Milkshape3D and Lightwave use different animation conventions, therefore a conversion to the Lightwave convention is done during import. This should work in most cases, but it is still possible that the animation conversion can fail and produce strange artifacts in the motion.

Milkshape3D note: Whenever you add a joint to the bone system in a milkshape model, the rest pose is recalculated. Therefore you should have added all the joints needed before importing and adding states to for instance a character. You cannot add states when the rest poses are different. For example: character_walk.ms3d and character_run.ms3d must have the same rest pose, otherwise the animation gets screwed up.

Good luck modding!