

ASSIMP Viewer Utility

Document version 1.0, April 2008

I. Usage

The ASSIMP Viewer Utility is a small and fast stand-alone viewer utility which is using the ASSIMP library to import assets from files. It consists of a single executable file and has no external dependencies. It displays assets with even highly complex materials and bone animations correctly.

AssimpView allows you to modify the textures and material properties sets of your models at runtime, easily and fast per Drag&Drop. Furthermore it can visualize normals, UV sets, bounding boxes ... It allows artists and engine programmers to work together with maximal efficiency and find the ideal model format for your internal workflow.

System requirements:

- A Direct3D 9.0c compliant video card with **at least** support for Shader Model 2.0.
- Shader Model 3.0 cards are recommended. Shader Model 2.0 cards always render the lighting in low-quality and have some limitations concerning complex materials.
- Windows 2000 or higher. AssimpView should also run on older versions, with DirectX 9 installed, but this has never been tested. If you're a proud owner of such a version and AssimpView works for you, let us know.

II. User interface

The viewer's user interface mainly consists of three components:

The menu bar provides access to the viewer's basic options.

- "**Viewer**": Items to load / unload assets, take screenshots and open the viewer's options dialog.

* "**Open Asset**": Displays the file system dialog where you can choose a new asset to load. Note that you can also Drag&Drop models onto the viewer panel.

* "**Close Asset**": Closes the current asset and releases all resources associated with it. No action if no asset is loaded.

- * "**Screenshot**": Save a snapshot of the viewer to a file. The snapshot does only include the preview panel, the rest of the user interface is not visible. Saved screenshots have no watermarks. This is Open Source Software!
 - * "**Reset View**": Reset the viewing position and direction to the default settings. This is: Camera position at 0|0|0, looking at 0|0|1.
 - * "**Memory consumption**": Display a box with approximated memory statistics. Note that the total memory consumption needn't be equal to the physical memory allocated to the process (as displayed by Windows Task Manager). The memory consumption does only refer to the memory required for the asset itself, not for the rest of the viewer.
 - * "**Setup file associations**": Associate all file formats which can be read by ASSIMP with the viewer. If you double click on such a file in Windows Explorer, it will be automatically opened with AssimpView.
 - * "**Recent files**": Displays a popup menu with a list of recently opened assets. Simply click on one to reload it.
 - * "**Clear history**": Clear the file history
 - * "**Quit**": What do you think?
 - "**Tools**": Additional utilities like stereo view, normal type.
 - * "**Log window**": Displays the output window of the logger. Warnings are displayed in orange/yellow, errors in red, debug messages in blue and status information messages in green.
 - * "**Save log to file**": Saves the contents of the log window to a file.
 - * "**Clear log**" Clear the contents of the log window
 - * "**Original normals**": Specifies that the original normal set from the model file is used in the preview. Normally this is the best choice, as smoothing groups etc. are handled correctly, but there are many models out there which have invalid or corrupt normal sets. However, if a model has no normal set, ASSIMP computes a smooth normal set for it. "Original normals" is in this case equivalent to "Smooth normals"
 - * "**Smooth normals**": Specifies that a smoothed, per-vertex, normal set is used in the preview window.
 - * "**Hard normals**": Specifies that a hard, per-face, normal set is used
 - * "**Flip normals**": Flip all normal vectors
 - "**Background**": Set the background color or use a texture (cube maps are supported as skyboxes) as background image for the viewer.
 - * "**Set color**": Displays a color picker where you can choose a new background color for the viewer.
 - * "**Load skybox**": Opens a file system dialog where you can select a skybox as background image. Accepted file formats are: *.dds, *.hdr.
 - * "**Load texture**": Opens a file system dialog where you can select a texture as background image. If the format of the texture doesn't fit to the proportions of the preview window, the texture will be stretched.
- Accepted file formats are: *.jpg;*.png;*.tif;*.tga;*.dds;*.hdr;*.ppm;*.bmp

- "?": Help and feedback

- * "**Feedback**": Opens a popup menu with two options:
- * "**Report bug**": Report a bug in AssimpView or in Assimp
- * "**Feature request**": Tell us what you'd like us to support. Any exotic file formats?

The side panel "*Statistics*" displays rendering statistics. The meanings of the fields change if you change between normal, node and texture or material view mode.

Normal view mode:

- * "**Verts**": Number of unique vertices in the asset
- * "**Faces**": Number of faces (= triangles) in the asset
- * "**Nodes**": Number of nodes in the scenegraph, including the root node
- * "**Mats**": Number of different materials found in the mesh. Most loaders remove unreferenced materials.
- * "**Mesh**": Number of sub meshes. Each mesh has only one material assigned, so this number is in most cases equal to or higher than the number of materials.
- * "**Shd**": Number of unique shaders created for this asset. If AssimpView detects that two materials can be rendered using the same Shader code, but with different parameterizations, it doesn't create the Shader twice. Most engines do so, too. This allows you to approximate the rendering const for the asset.
- * "**Time**": Time required for loading, in seconds. This time is the raw time the process spent in Assimp loading the model itself, it does not include texture loading, Shader compilation...

Texture view mode:

- * "**Width**": Width of the texture, in pixels
- * "**Height**": Height of the texture in pixels
- * "**Format**": Format of the texture. This will normally be ARGB8. Other possible values are: RRG565, ARGB16f, ARGB32F. Embedded textures are always converted to ARGB8 format, even if they're stored in the file with a higher color depth.
- * "**UV**": UV channel used by the texture. Normally this is the first channel.
- * "**MIPs**": Number of MIP map levels created for the texture
- * "**Blend**": Strength of the texture. The color value obtained from the texture is multiplied with this factor to calculate the final color.
- * "**Op**": Blend operation used to combine this texture with the corresponding material color (or the previous texture).

Always visible:

- * "**FPS**": Current frame rate, in frames per second. Note that this number is not really exact. It is only an approximation to the real frame rate. The maximum frame rate will never be reached since there is a 10 ms blocker included (to work around good our old AMD timing bug)

The *“Rendering”* panel provides rendering-related options, such as wireframe/normal visualization.

“Interaction” bundles all input related options.

The main component is the large viewer panel where the assets are displayed. Status messages (e.g. if a texture could not be loaded) are displayed in the upper-right edge. Yellow messages are simple information messages, red messages are error messages. Green messages, however, are coming from heart ;-)

III. Input

The input mode depends on the selected input behavior. If *„Zoom/Rotate”* is **NOT** selected the following scheme applies:

- ‘Arrow up’* = Move forwards
- ‘Arrow left’* = Move to the left
- ‘Arrow down’* = Move backwards
- ‘Arrow right’* = Move to the right

The mouse specifies the view direction. This is the typical FPS input behavior.

Otherwise, if *„Zoom/Rotate”* is enabled:

“Left Mouse Button” = Keep pressed and move the mouse to rotate the asset around its local axes. Inside the yellow circle: x/y-axis, outside: Z-axis. To rotate around one axis only use the axis snap-ins (yellow squares with a cross inside).

“Right Mouse Button” = Keep pressed and move the mouse to rotate the light source(s) around their x and y axes.

“Middle Mouse Button” = Keep pressed and move the mouse from the left to the right or the other way round to increase/decrease the intensity of the light source(s)

*“Right **AND** Left Mouse Button”* = Keep pressed and move the mouse to rotate skybox (if existing) around the global x/y axes.

“Mouse wheel” = Zoom in/out

IV. Rendering

Enabling MultiSampling improves rendering quality. MultiSampling is activated by default. The highest quality MultiSampling mode supported by the video card is used.

MultiSampling is especially useful to remove line artifacts in wireframe/normals mode. Note that the transition between normal and multisampled rendering may take a few seconds.

Rendering is done via Direct3D™ 9.0c. Note that the "*Low-quality lighting*"-Option is not available for PS 2.0 cards. Lighting on PS 2.0 cards is always low quality.

v. Known issues

- If a loading process is canceled it is not guaranteed that further loading processes will succeed. ASSIMP Viewer might even crash in this case.
- Some very complex materials involving real time reflection/refraction are not displayed properly.
- When rendering non-opaque objects some triangle artifacts might occur.