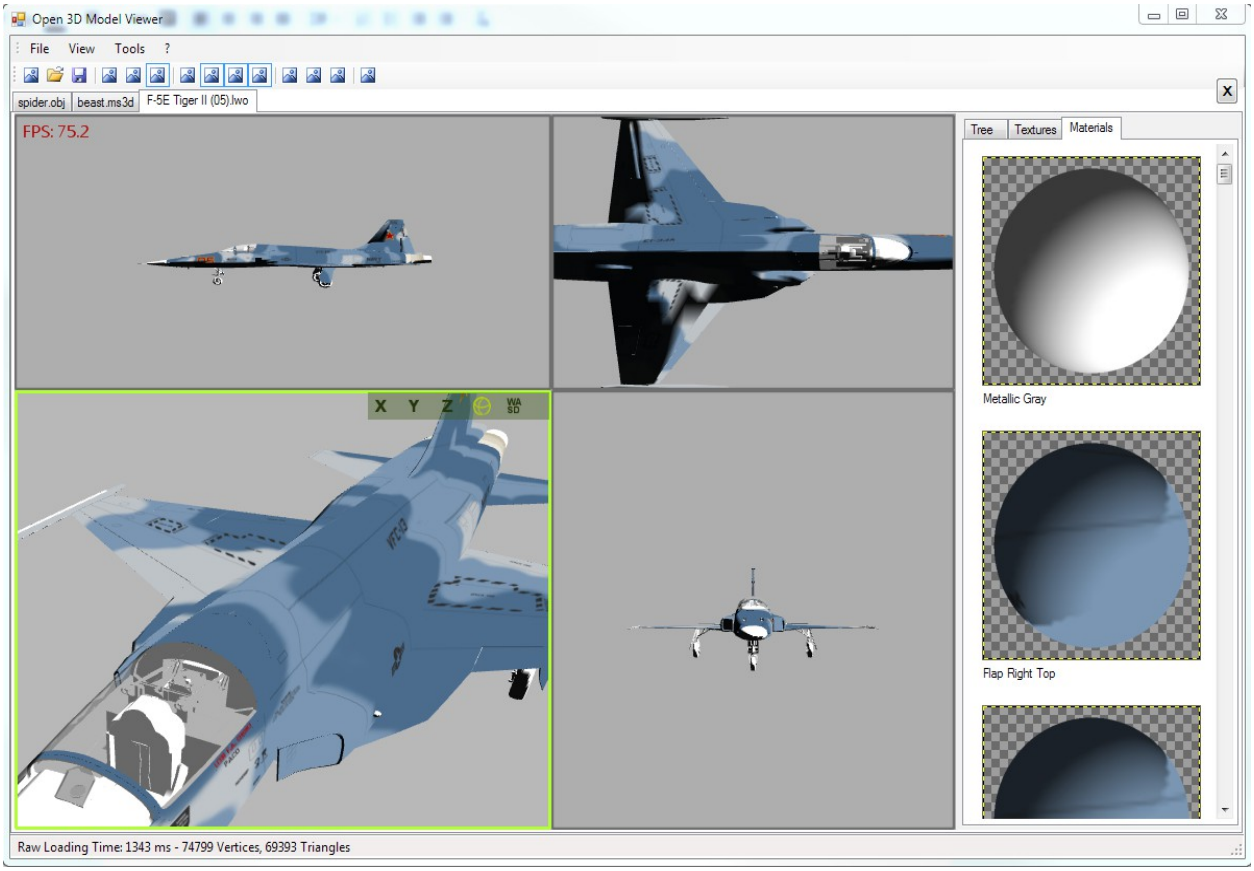
Design Project: Icons for 3D Viewer Application

Scope

Open3mod is an open-source (that is, free!) 3D model viewer that is currently in development. Even though it is offered without charge, its style should be professional, clean and have an intuitive UI. **To achieve this, I need a set of totally 19 icons.**

It would be easier for you, if you have some experience with 3D terminology, but this is not required. The work-in-progress version of open3mod looks like this:



Target Users

The application will be used by hobbyist and professional artists who work with 3D data, as well as by software developers working on 3D graphics projects

---

Deliverables

16 icons with 16x16 px plus 3 icons with 24x24 px. All in PNG format with proper transparency.

Overall aesthetic

Might be helpful to know that I'm a computer scientist who targets Windows users with this project. So, as for the icons, see image below for the style I would like to have emulated but not directly copied, as these are from Microsoft Visual Studio 2012 respectively Microsoft's overall "Metro" style. However, any other style you come up with is appreciated too - I'm open to your creativity!

Some guidelines:

- simple, not too detailed
- no shading
- not too many colors per icon

Part 1: Tool Bar

The following is the list of icons used in the main tool bar, followed by a description the corresponding UI action.

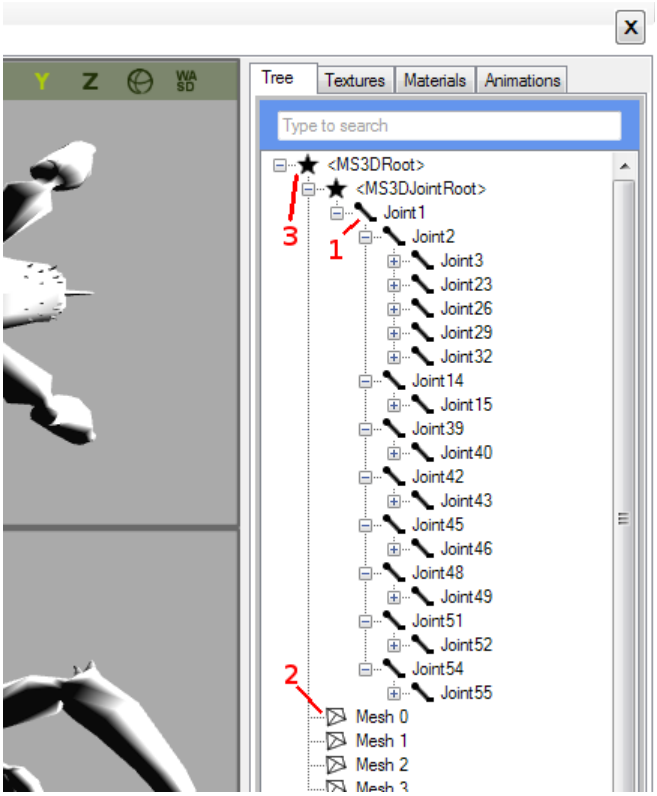
Needed: **12 Icons in PNG format (with transparency), 16x16 px**

Icon List:

1. *Load* - Opens a file-open dialog where the user can select a 3D file to be opened
2. *One View* - Sets the viewer to have only one 3D view
3. *Two Views* - Sets the viewer to split the 3D view vertically
4. *Four Views* - Sets the viewer to split the 3D view in four parts (as in the screenshot)
5. *Bounding Boxes* - Sets the viewer to show bounding boxes for objects
6. *Normals* - Sets the viewer to show normal vectors for meshes (easy to illustrate by an arrow perpendicular to a surface)
7. *Skeleton* - Sets the viewer to show the animation skeleton that underlies the model. Maybe a "skeletal stick figure" could be used to illustrate this.
8. *Shading* - Icon for a checkbox that specifies whether the 3D preview is lit (light and shadow) or not (all full brightness).
9. *Textured* - Icon for a checkbox that specifies whether the 3D preview shows texture images or not.

10. *Wireframe* - Icon for a checkbox that specifies whether the 3D preview is in wireframe or in solid drawing mode. In Wireframe mode, the outlines of triangles are drawn, not the interior (i.e. just lines).
11. *Settings* - Opens the settings dialog
12. *Export* - Opens the export dialog, which offers to export the scene to one of several 3D formats.

Part 2: Scene Structure View



The Scene Structure View outlines the (hierarchical) structure of the 3D model being shown. The icons for it will replace the not-so awesome placeholders on the screenshot.

Needed: **4 Icons in PNG format (with transparency), 16x16 px**

Icon List:

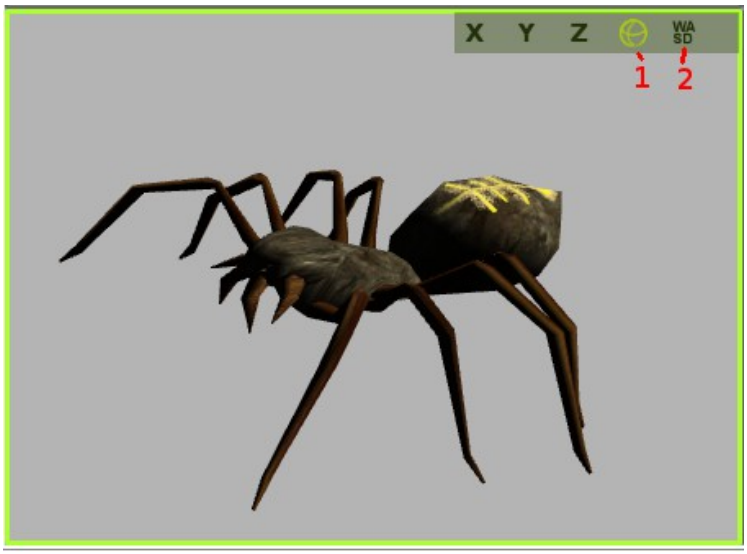
1. *Joint* - represents a skeleton joint
2. *Mesh* - represents a mesh (i.e. some triangular geometry)
3. *Root* - represents dummy nodes introduced for technical reasons
4. *Normal* - represents anything else (not shown in the shot on the right, a very simple illustration is ok for this, the placeholder is just a black filled circle)

Part 3: 3D View Overlay

This UI element act as an overlay over the 3D view when the user hovers over it. It allows the user to change camera modes or set per-view settings. It is shown against a half-transparent, gray background, which eventually will be replaced by a slight gradient starting at the corner. The icons on the overlay should be black/grayscale. There will be a “hover” and a “selected” state, but this is achieved by color-tinting the images (I will do this).

The X,Y,Z view modes which can be seen on the screenshot don’t need new icons.

Needed: **3 Icons in PNG format (grayscale, with transparency), 24x24 px**



Icon List:

1. *Orbit view* - in this mode, the user can rotate the scene by moving the mouse and scroll using the mouse wheel (i.e. the camera is locked on an orbit around the scene). The placeholder is supposed to be a globe with arrows in both axes, but anything that illustrates the concept is ok.

2. *First-Person-View* - in this mode, the user gets the classic WASD-controls known from shooters.

3. *Picking-View* - (not seen in the screenshot) in this mode, the scene is locked and the user gets to pick elements and see basic information on them.