
Subject: Mesh software kickstarter
Posted by [henryseg](#) on Thu, 15 Nov 2012 10:27:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

MeshUp: Mashup for meshes is a kickstarter for a "Super simple, always watertight, 3D modeling mashup tool for meshes, designed for painless and direct 3D printing."

I hadn't heard of the company before, but it looks useful!

Subject: Re: Mesh software kickstarter
Posted by [aeron203](#) on Thu, 15 Nov 2012 16:45:36 GMT
[View Forum Message](#) <> [Reply to Message](#)

MeshUp looks very promising.

I wrote a very in-depth blog post analyzing the problems it intends to solve, what it can and can't do, and comparing it to other solutions here:

Uformia's Volumetric Solution

Subject: Re: Mesh software kickstarter
Posted by [henryseg](#) on Thu, 15 Nov 2012 22:57:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks Aaron, that's very informative. It sounds like it will solve many of the kinds of problems I run into.

Subject: Re: Mesh software kickstarter
Posted by [AlanHudson](#) on Fri, 16 Nov 2012 18:04:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Interesting read on uformia and your triangle / voxels article. I've been working on an opensource codebase called AbFab3D that is a server-side(Java) volumetric toolkit. It powers a few features in our backend currently.

One of the things I hate is we only take triangle input. During the processing we create voxel

representations to perform some analysis. But we have no path to transmit voxel models to printers.

If we had a good voxel format then we might be able to stay in voxels the whole way. Do you know of a good voxel transmittal format? Really only DICOM comes to mind and its a huge standard. I've been involved in a bunch of 3D triangle format definitions(VRML,X3D,COLLADA) so its possible I might lead a voxel definition effort if necessary.

AbFab3D will be making its 1.0 release milestone soon. We've got some nice isosurface extraction tools now that make it useful to work in voxel space and then output to triangles. Check out this model if you want to see some output from it:

<http://www.shapeways.com/model/730555?li=wishlist&materialid=37>

I'll be making more noise about AbFab3D soon once I have time to cleanup the website and code for its 1.0 debut.

Subject: Re: Mesh software kickstarter
Posted by [aeron203](#) on Fri, 16 Nov 2012 19:45:42 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Alan,

I remember taking a look at AbFab3D, it looks very useful, though making it part of a framework for a full featured design package rather than a Creator is putting a huge amount of work on somebodys plate.

I have dealt with voxel storage and transmission to hardware with my previous work with 3D displays, we used a plain RLE. This is very common since even the simplest microcontroller can process it easily. We abandoned DICOM for storage almost immediately due to the patent. It is really a TIFF sequence +ancillary medical data, and I think an open standard should not be hard to make from scratch. I know about the plans to expand AMF to include voxels, but as much as I want that to work, there is an obvious problem- the data would be MIME encoded. Each binary voxel would occupy two characters, 16 bits, on disk. Using multiple materials requires a separate copy of the data for each material (!). This structure would not be useful. Even if it's compressed, since it's all in one file, the printer would probably need enough memory to store the whole unpacked file, and would have to chew through the whole thing to access some arbitrary portion of it. So, any printer manufacturer claiming to support AMF will need to pack in a bloated multi-layer stack of applications, multi-core processing, hundreds of GPU's, and at least a Terabyte of solid state memory. That will be one expensive and complicated piece of equipment for no good reason. Half a century of CNC development has taught us to keep the output machine dumb and do the high-level work in the design stage.

Many voxelizer's output RLE blocks, but I am not a fan of using interleaved slices. They should be separate and human readable. In my own work I use PNG sequence (often in an AVI wrapper) because it has the best compression and supports 1, 4, 8,16, 24, 32, and 64 bit depths, which leaves room for lossless storage of high-dynamic range data or indexed multi-material models. Since PNG is already open and included in most image processing libraries on every platform, it is easy to implement. It is harder to unpack using a microcontroller, but a Raspberry Pi or something similar could do the job. I anticipate even more integrated ARM-based systems that more closely resemble the guts of a smartphone to take over this space very shortly.

Obviously the function of MeshUp and AbFab3D are totally different, since is design software and one is a framework, but they are addressing different ends of the same problem, and the best thing that could happen would be for a common standard to be adopted by everyone working on these issues.

Here is how I would like to see this develop in approximate order:

1. AbFab3D toolkit (and subsequently Shapeways) support model input as PNG sequence with separate text header.
2. Online tools of various flavors are developed using AbFab3D kit to do basic processing and generate the appropriate header by letting the user upload PNG sequence and enter parameters (resolution, materials, iso thresholds, etc.)
3. Uformia adopts this standard so SymVol and MeshUp can convert and output compatible files.
4. Sculpting software companies like Pixologic's Zbrush adopt polygon free output. Autodesk has shown their openness in this area, and would probably add support in 123D sculpt and MudBox within a year or so.
4. Geomagic integrates the format into the Sensable haptic sculpting systems (Claytools and FreeForm), and Geomagic proper.
5. Mesh fixing tools like Magics and Netfabb would also benefit from robust volumetric processing.
6. Additional tags created for AMF and perhaps VRML/Collada to enable combined voxel/mesh models by referencing the external image sequence in a manner similar to textures in VRML/Collada. Hopefully AMF will adopt this officially, but if not, let's do it anyway.
7. Gain compatibility from voxel-based DIY printer makers who are more open to adopting new standards. B9creator, Form1, Blu-Printer are main candidates.

8. Feeling left behind, Zcorp and Objet, and EOS will adopt the standard.
9. A complete industrial-scale, polygon free model pipeline is established.
10. Everyone profits.

Subject: Re: Mesh software kickstarter
Posted by [AlanHudson](#) on Fri, 16 Nov 2012 21:36:22 GMT
[View Forum Message](#) <> [Reply to Message](#)

"Hi Alan,

I remember taking a look at AbFab3D, it looks very useful, though making it part of a framework for a full featured design package rather than a Creator is putting a huge amount of work on somebodies plate."

Agreed, hoping someone else does it, hence why its open source. I've been working on more full featured creators recently so its getting possible but all the user interface will take a long time.

"I have dealt with voxel storage and transmission to hardware with my previous work with 3D displays, we used a plain RLE. This is very common since even the simplest microcontroller can process it easily. We abandoned DICOM for storage almost immediately due to the patent. It is really a TIFF sequence +ancillary medical data, and I think an open standard should not be hard to make from scratch. I know about the plans to expand AMF to include voxels, but as much as I want that to work, there is an obvious problem- the data would be MIME encoded. Each binary voxel would occupy two characters, 16 bits, on disk. Using multiple materials requires a separate copy of the data for each material (!). This structure would not be useful. Even if it's compressed, since it's all in one file, the printer would probably need enough memory to store the whole unpacked file, and would have to chew through the whole thing to access some arbitrary portion of it. So, any printer manufacturer claiming to support AMF will need to pack in a bloated multi-layer stack of applications, multi-core processing, hundreds of GPU's, and at least a Terabyte of solid state memory. That will be one expensive and complicated piece of equipment for no good reason. Half a century of CNC development has taught us to keep the output machine dumb and do the high-level work in the design stage."

One thing I discovered about STL's design was that it was streamable. No need to keep it in memory. No forward/backward references. It was an epiphany of sorts since I've been so used to working on full featured formats like X3D/COLLADA. If you really want to handle large files you need to design the format for streaming.

"Many voxelizer's output RLE blocks, but I am not a fan of using interleaved slices. They should be separate and human readable. In my own work I use PNG sequence (often in an AVI wrapper) because it has the best compression and supports 1, 4, 8,16, 24, 32, and 64 bit depths, which leaves room for lossless storage of high-dynamic range data or indexed multi-material models. Since PNG is already open and included in most image processing libraries on every platform, it is easy to implement. It is harder to unpack using a microcontroller, but a Raspberry Pi or something similar could do the job. I anticipate even more integrated ARM-based systems that more closely resemble the guts of a smartphone to take over this space very shortly."

64 bit might be enough. For the best isosurface extraction you'd like to carry around a normal and you'll want the material. So it's close. I'd agree that a stack of PNG's would be a good choice. When I was working with NASA they were pretty interested in this format:
<http://en.wikipedia.org/wiki/NetCDF>

Have any experience with it? It's a possible candidate but I suspect more full featured than we need. But it does have the creds of handling really large datasets.

"1. AbFab3D toolkit (and subsequently Shapeways) support model input as PNG sequence with separate text header."

How much control would you need over the isosurface extraction? We've got a decent algorithm going but we have found we like to Laplacian smooth the results to decrease stair stepping. So not sure if this step could be speced well enough to be standardized between implementors. I feel there is some art here and just using marching cubes might not be acceptable.

What things would you expect in the header? I'm mostly thinking an index of pngs, the number of them, grid / world conversion , grid resolution.

"6. Additional tags created for AMF and perhaps VRML/Collada to enable combined voxel/mesh models by referencing the external image sequence in a manner similar to textures in VRML/Collada. Hopefully AMF will adopt this officially, but if not, let's do it anyway."

I worked on the Volume Rendering component for X3D and it generally solved the problem as a stack of images so it would fit pretty well into this. If we just supported the VolumeData node and a stack of png's that might be enough.

Care to take a look at that definition and see if it would meet your needs?

<http://www.web3d.org/files/specifications/19775-1/V3.3/index.html>

It's a rendering standard so the concept of materials is not really well covered. Maybe we have a

PhysicalMaterialPrintingStyle or something to denote real physical properties.