

mk_L3DTio_DXStudio: A DXStudio DXMesh Exporter Plugin for L3DT **v3**

Content

Versions	1
Features.....	2
Installation.....	2
Exporter Dialog.....	4
Status Dialog.....	5
Central XML info file	6
DXStudio Demo Doc	7
Usage Example1: Small Map Single Mesh Export	8
Usage Example 2: Huge Map/Tiled Mesh Export.....	11
Issues/Bugs.....	14
Contact/Download	14
Credits	14

Versions

1.0.0.1	6/2011	first BETA release, with Debug output to e.g. <i>DebugView</i>
---------	--------	--

Features

This plugin lets you export a L3DT terrain as DXStudio DXMesh.

- One single mesh or tiled terrain (“mosaic”)
- Automatic export of vertex normals for the mesh (using L3DT-internal calculation routines)
- Export mesh with texture maps (if present in L3DT)
- Export with “tangent space” normal maps (if present in L3DT; this is a L3DT **Pro** feature)
- Export with or without face count reduction (using L3DT-internal calculation routines)

On export, a central XML file is created. This can be used to position terrain tiles at runtime. Also, it contains some additional information (like face count per mesh, etc).

On export, a DXStudio demo doc is created (DXStudio player installation required). By double-clicking this file, it will load and show the terrain(s) in the same directory, using DXStudio’s built-in background loading.

Important notes:

- This plugin is for L3DT **v3**. It was tested with a preview dev version, and is supposed to work with L3DT v2.9.0.22 (developmental build) and up.
- Although tiled export does work for L3DT Std and Pro, the restrictions with the standard version (especially: texture size) do not really produce “great” results. It is strongly recommended you go for the Pro version of L3DT (definitely worth it, anyway).

Installation

The installation download comes with two files:

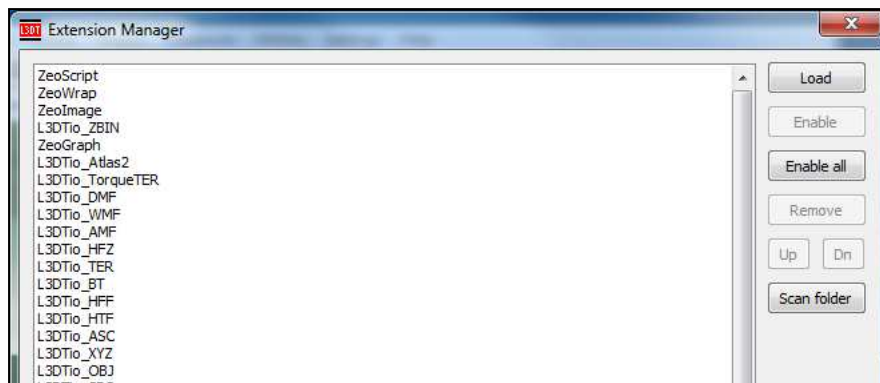
- mk_L3DTio_DXStudio.zeo
- DXMeshAPI.dll

The *.dll is a DXMesh helper (“DXMesh API”), provided by Worldweaver, the creators of DXStudio. This one is utilized by the *.zeo file (which is in fact the export plugin dll for L3DT).

Copy these two files into the L3DT “Extensions” folder. The destination could something like

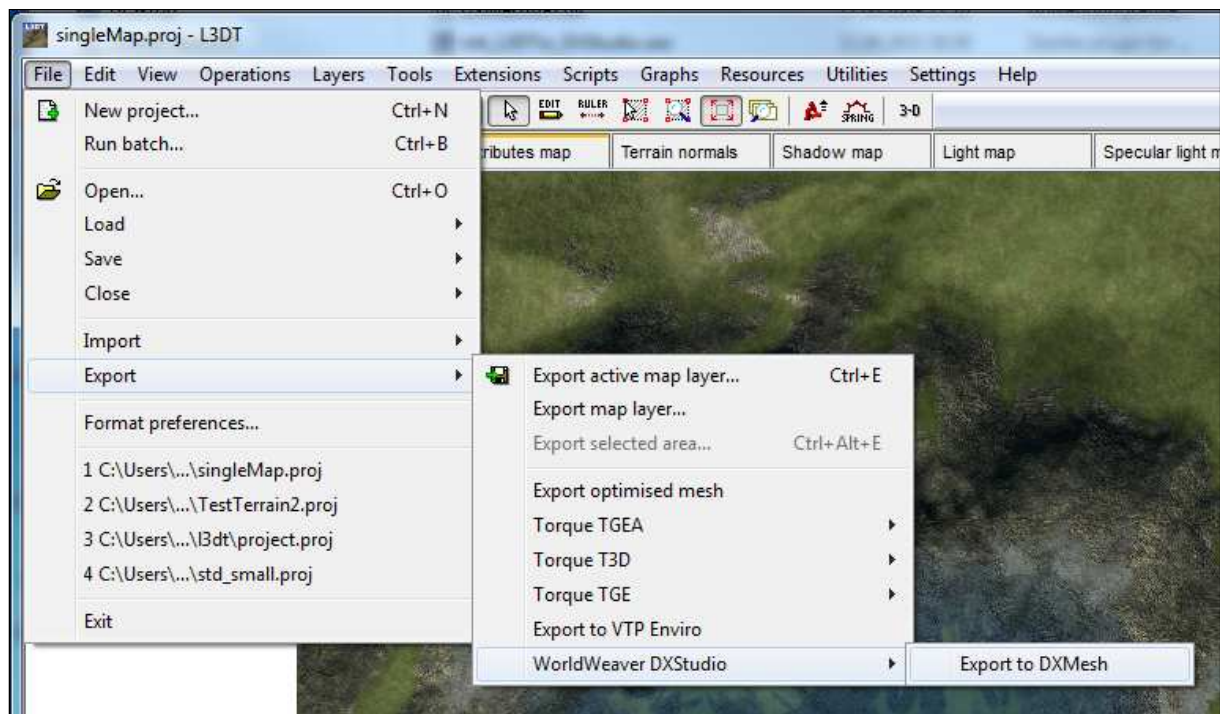
C:\Program Files (x86)\Bundysoft\L3DT Professional 2.9.0.22\Extensions or
C:\Program Files (x86)\Bundysoft\L3DT Standard 2.9.0.22\Extensions

After that, you need to enable the new plugin within L3DT. To do so, open the “Extension Manager” (Menu -> Extensions), select “Scan Folder”, choose the “Extensions” folder and run. A popup window will inform you about new plugins found and ask for installation.



Press “Yes” for *mk_L3DTio_DXStudio* and close the Extension Manager (the program will alert you that it needs to restart).

After successful installation, a new menu option is available in the L3DT “File” menu section.



Exporter Dialog

“Map Info”

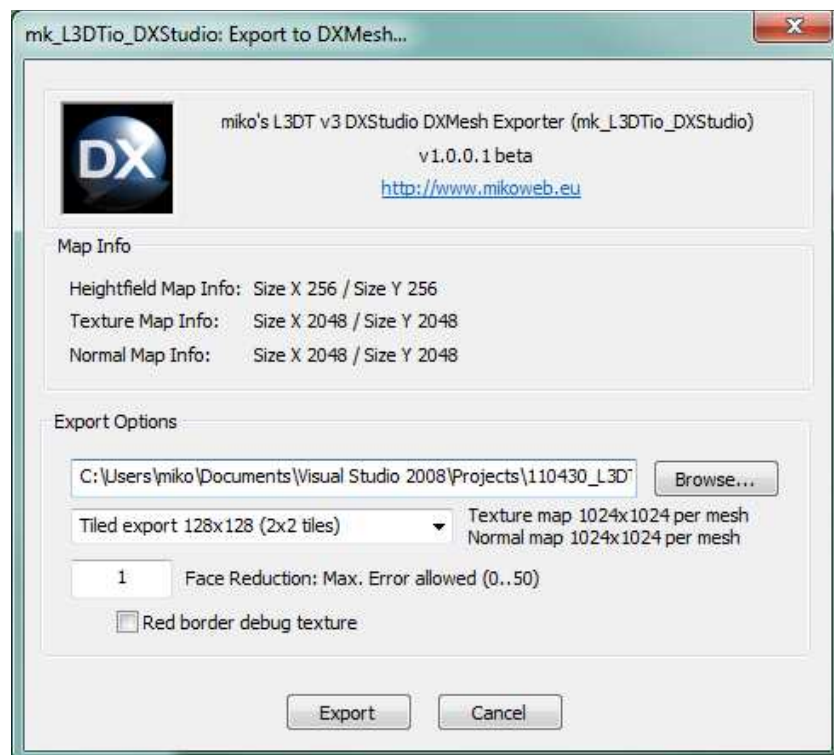
The exporter dialog (that comes up when invoking the plugin via the “File” menu) shows you some info about the current height field, texture map and normal map. The height field is mandatory (you can’t proceed without it), while the texture/normal maps are optional. When present, they will be exported.

“Export Options”

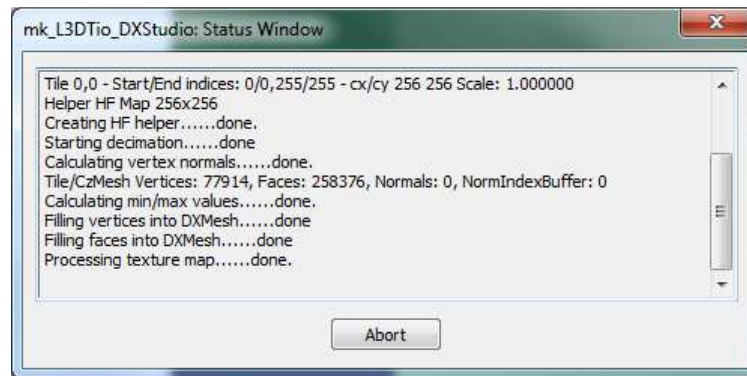
Here, you need to choose the location where to store the new files (preferably, use a dedicated folder). Also, you can select “tiled” (=mosaic) or single mesh export. The corresponding size of the texture/normal map per mesh will be updated on change.

“Face Reduction Max. Error” uses L3DT built-in routines to reduce the number of faces on export. Possible values here are 0(=no reduction) up to 50 units (integers only). Note that you can check the actual face count of each mesh in the *.xml info file, created on export. For now, it is not possible to do that interactive and “on-line”.

For debug purposes in your application (like, positioning terrain), the exporter can create terrain meshes with a red bordered texture.



Status Dialog



Once the export is running, the current process step will be shown in the status dialog. Be aware that some operations can take a long time! The dialog will automatically close when the export is finished. By pressing "Abort", the export will be stopped *after* the current process (so, it might take a while for it to respond).

Central XML info file

With each export, a *.xml that contains terrain info is created. An example:

```
<TerrainInfo>
  <General>
    <Tiles x="2" z="2" />
    <HF_TileSize xz="128" />
    <TX_TileSize xz="1024" />
    <TN_TileSize xz="1024" />
  </General>
  <TileInfo>
    <Tile x="0" z="0" fPosX="0.000000" fPosY="0.000000" fPosZ="0.000000"
vertices="19921" faces="196488"/>
    <Tile x="1" z="0" fPosX="128.000000" fPosY="0.000000" fPosZ="0.000000"
vertices="19747" faces="195588"/>
    <Tile x="0" z="1" fPosX="0.000000" fPosY="0.000000" fPosZ="128.000000"
vertices="19724" faces="195204"/>
    <Tile x="1" z="1" fPosX="128.000000" fPosY="0.000000" fPosZ="128.000000"
vertices="19608" faces="195318"/>
  </TileInfo>
</TerrainInfo>
```

In the *General* section, it shows the number of tiles (x,z), the terrain tile size (HF) in units, the texture (TX) size per tile (in pixels) and the normal map (TN) size in pixels.

For each exported tile, the recommended position in DXStudio world space (X/Y/Z) is listed, as well as the number of vertices and faces. Note that the pivot point (0/0/0) for any terrain tile (may it be single or mosaic) is always at its lower left corner.

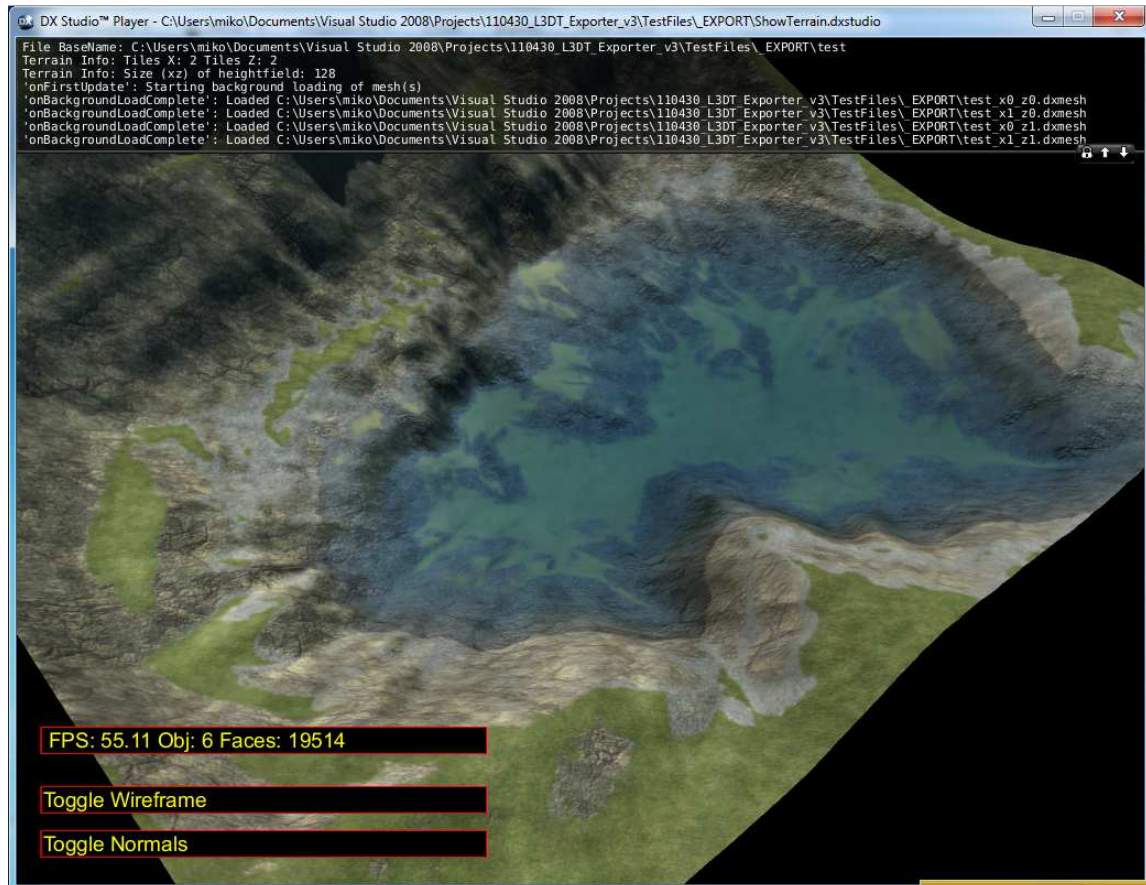
Tterrain meshes are always created with an additional index suffix like “_x0_z1”. This does apply for single mesh export aswell. So, in case you entered an export name of “myTerrain”, a finally created mesh would be named like “myTerrain_x0_z0”.

The *.xml info file can be loaded by a DXStudio application (“doc”) and be used for easier placing of terrain tiles (as shown in “ShowTerrain.dxstudio”, which is generated on export, too).

DXStudio Demo Doc

With each export, a DXStudio demo doc (named “*ShowTerrain.dxstudio*”) is created. By double-clicking it (DXStudio player must be installed), it will load the terrain(s) in the current directory and display them for a fast check. You can toggle wireframe mode and show vertex normals, too.

This doc does also show how one could easily load terrain tiles using the provided *.xml file (for single-mesh terrains, you could still use the DXStudio Mesh Viewer, of course).

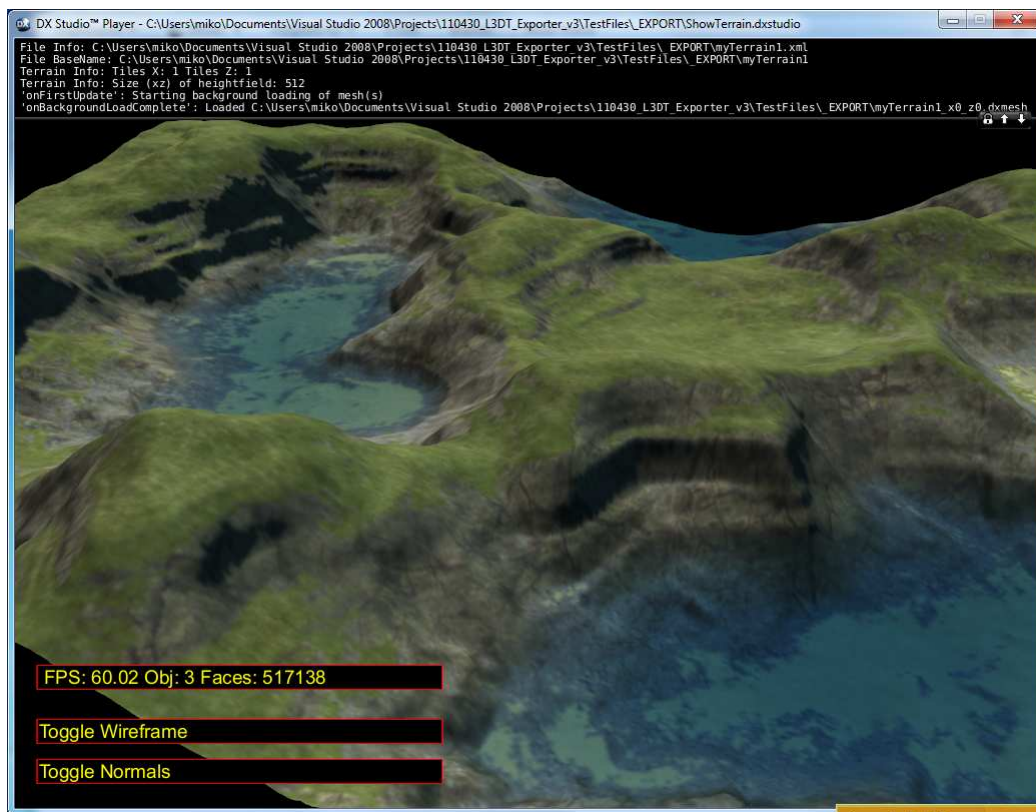


Usage Example1: Small Map Single Mesh Export

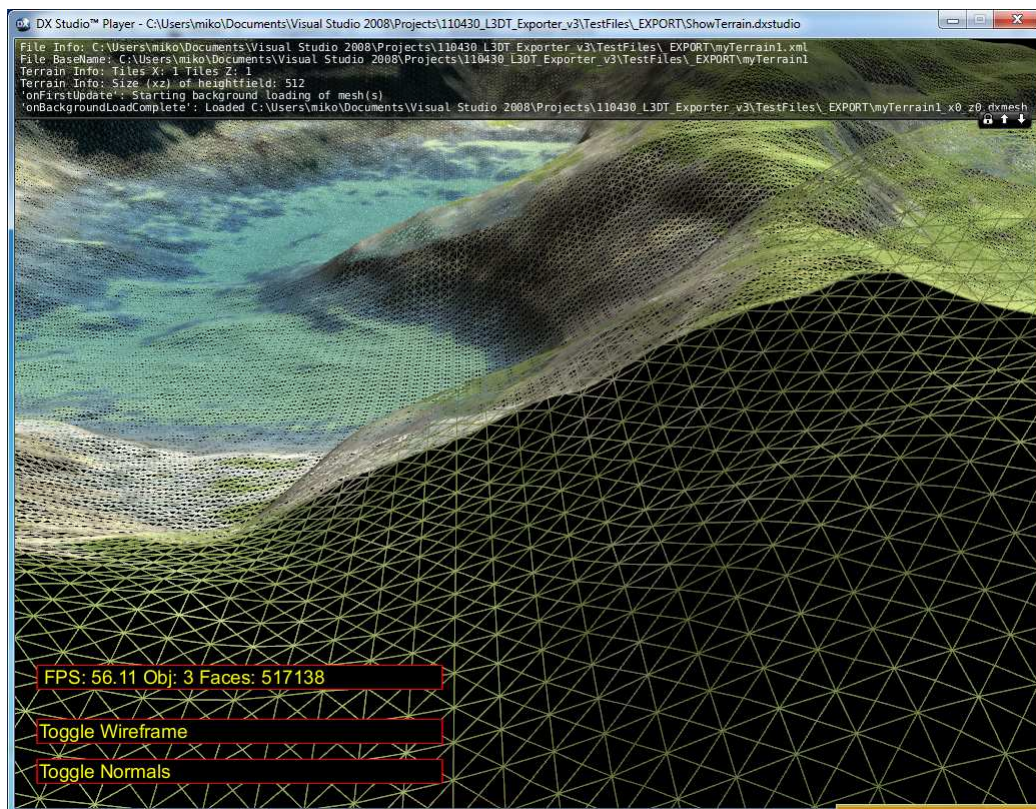
For this example, we'll use the L3DT **Std** version. Goal is to create and export a single DXStudio terrain mesh (facecount not reduced), fully textured and – as tangent normals are L3DT Pro only – not normal mapped.

- Install and activate the export plugin as described above.
- Create a new project (“Designable map”) **Next>>**
- Choose “Width” and “Height” of 512 each
- Select a “Horiz. Scale (m)” of 1, to make the terrain 512x512 units big in DXStudio
- Leave the two check boxes unchecked **Next>>**
- Leave the HF/DM ratio as presented **Next>>**
- Leave the Design map parameters as presented **Next>>**
- In the calculation queue, activate *all* checkboxes, so we can do in one run **Next>>**
- Leave the Water flooding options as presented **Next>>**
- Leave the Water-table modelling as presented **Next>>**
- Leave the Light mapping(1) as presented **Next>>**
- Leave the Light mapping(2) as presented **Next>>**
- Leave Light/water effects as presented **Next>>**
- In Texture Settings, check “Make high-resolution texture” and select “1024x1024” (the limit for the L3DT Std would be 2048x2048 here) **OK>>**
- Save the project if requested by L3DT
- Watch all the magic calculations to happen
- Now, invoke the DXMesh exporter dialog as described above
- Press “Browse” and select an empty folder (where the terrain will be exported to) and enter a name, e.g. “myTerrain1”. Close the “file save as...” dialog.
- Leave “No tiled export” selected and “Face Reduction” at 0
- Press “Export” and wait for the progress dialog to close
- Now, open the folder where the terrain is stored and double click “ShowTerrain.dxstudio” (it might take a while for the mesh to be visible, as it is loaded in background)
- Use mouse/WASD to move the camera around. Toggle “Wireframe” and “Normals” for more info (showing the Normals might cause heavy load and slow response)
- Done

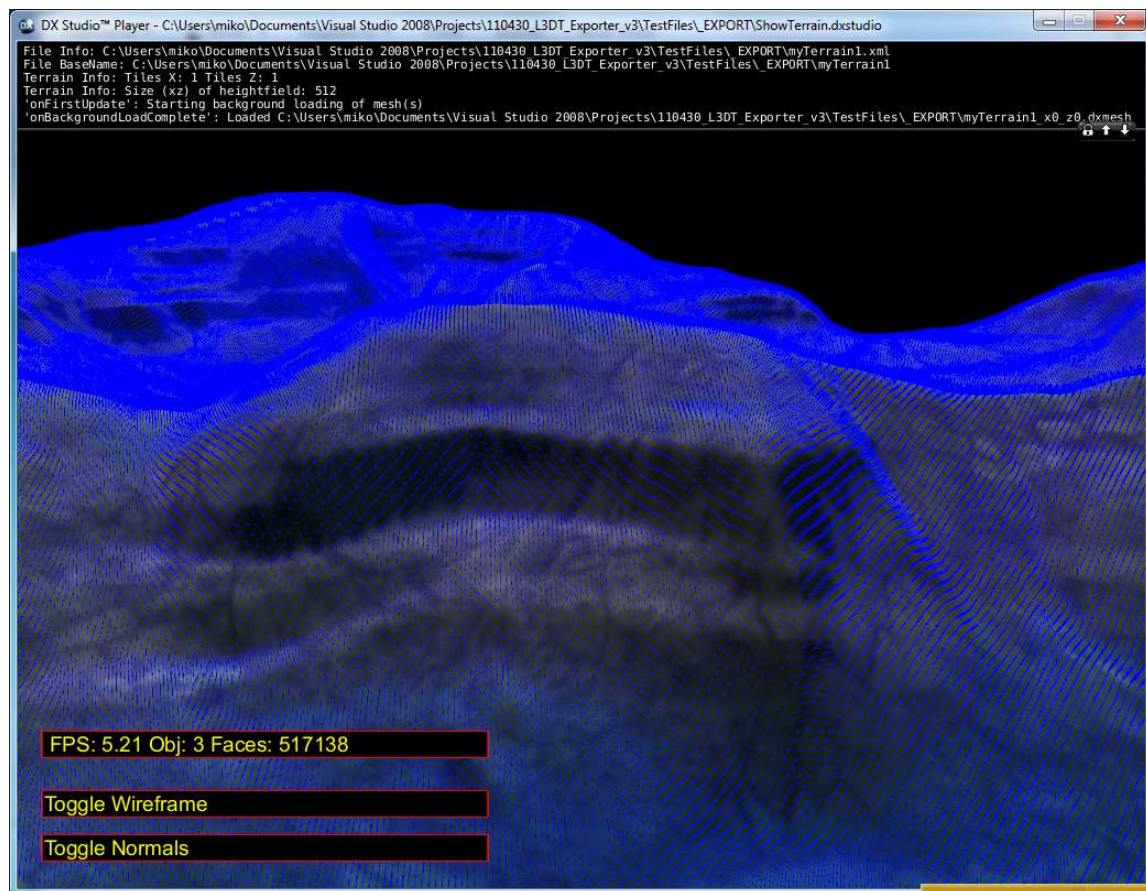
The created terrain in the demo doc



Wireframe mode



Vertex Normal indicators switched on



Usage Example 2: Huge Map/Tiled Mesh Export

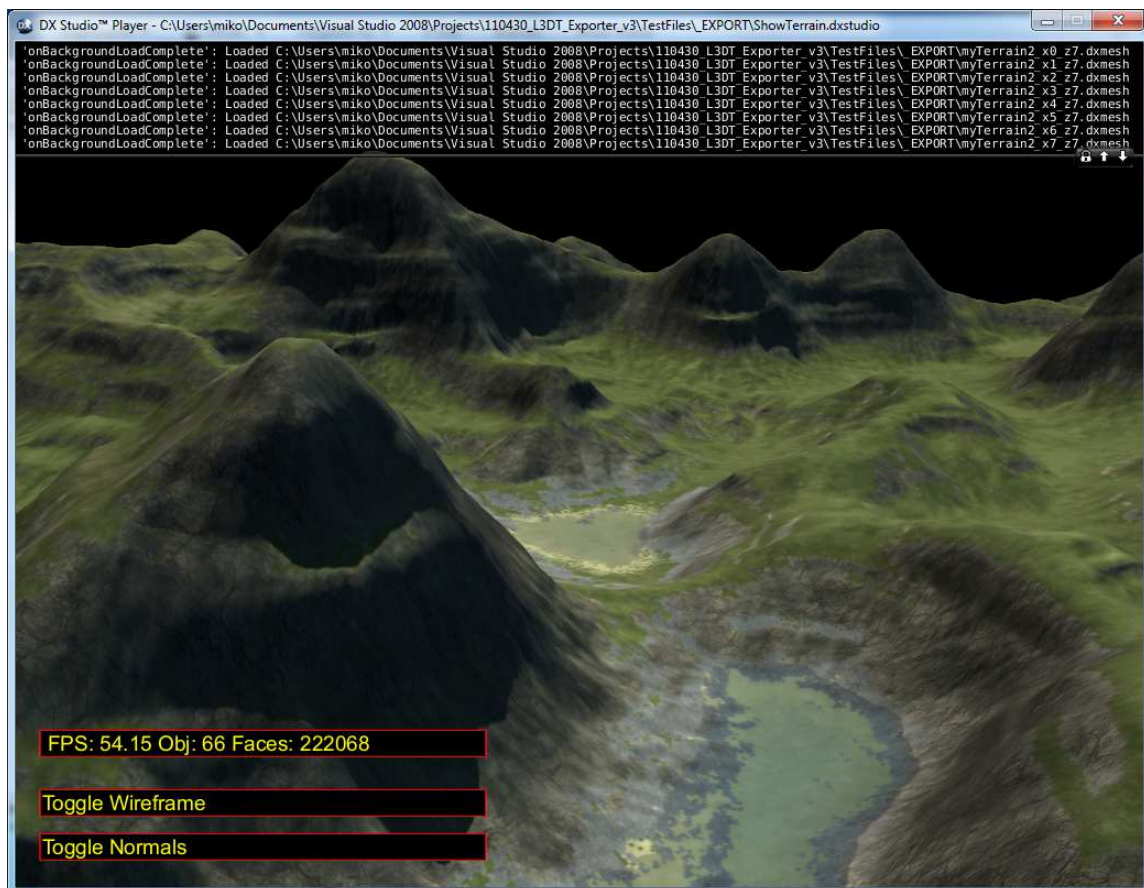
For this example, we'll use the L3DT **Pro** version. Target is to create and export a tiled DXStudio terrain mesh ("mosaic terrain"), fully textured and normal mapped in tangent space. Also, the face count will be reduced.

Let say we want to make a 8x8 tiled terrain (64 tiles), with 1024x1024px texture/normal maps per tile. The size of each mesh tile shall be 128x128 units, so we have a considerably small land area covered by a considerably large texture.

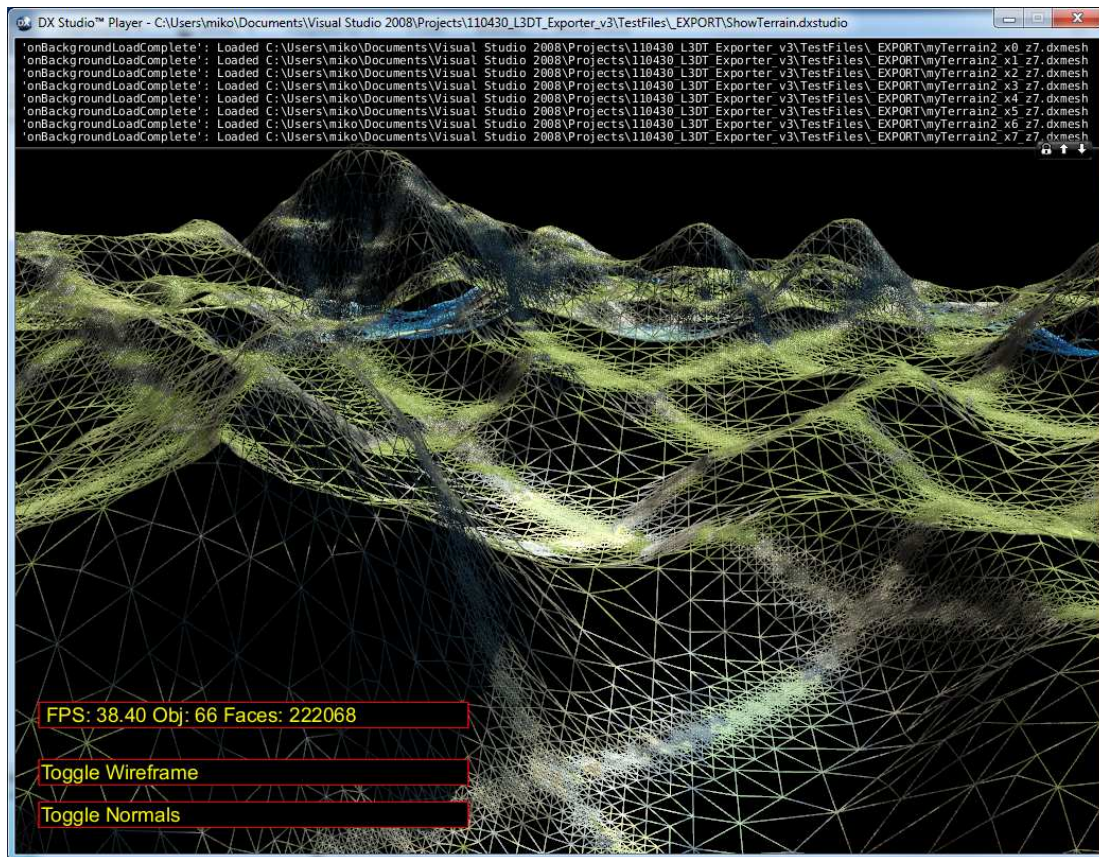
- Install and activate the export plugin as described above.
- Create a new project ("Designable map") **Next>>**
- Choose "Width" and "Height" of 1024 each
- Select a "Horiz. Scale (m)" of 1, to make the terrain 1024x1024 units big in DXStudio
- Leave the two check boxes unchecked (the exporter will split by itself) **Next>>**
- Leave the HF/DM ratio as presented **Next>>**
- Leave the Design map parameters as presented **Next>>**
- In the calculation queue, activate *all* checkboxes, so we can do in one run **Next>>**
- Leave the Water flooding options as presented **Next>>**
- Leave the Water-table modelling as presented **Next>>**
- Leave the Attributes map calc. as presented **Next>>**
- In "Normal Mapping", activate "Apply bump mapping" and "Generate tangent-space bump map layer". This layer will be used as the meshes' normal map.
Also, activate "Make high-res normals map" and adjust it to 8192x8192 pixels (as we have 8x8 tiles finally, each tile will thus have a 1024x1024 texture) **Next>>**
- Leave the Light mapping(1) as presented **Next>>**
- In Light mapping(2), activate "Make high resolution light map" and adjust it to 8192x8192 pixels (as we have 8x8 tiles finally, each tile will thus have a 1024x1024 texture) **Next>>**
- Leave Light/water effects as presented **Next>>**
- In Texture Settings, check "Make high-resolution texture" and select 8192x8192 pixels (as we have 8x8 tiles finally, each tile will thus have a 1024x1024 texture) **OK>>**
- Save the project if requested by L3DT
- Watch all the magic calculations to happen. Now, this will take a long time here. **snooze*
- When finished, invoke the DXMesh exporter dialog as described above
- Press "Browse" and select an empty folder (where the terrain will be exported to) and enter a name, e.g. "myTerrain2". Close the "file save as..." dialog.
- Select "Tiled export 128x128 (8x8 tiles)" and set "Face Reduction" to 2 units (adjusting the reduction value might take several time consuming exporter runs – which is inconvenient, I admit)
- Press "Export" and wait for the progress dialog to close. Again, this will take some time, as 64 terrain tiles need to be created. **snooze*
- Now, open the folder where the terrain is stored.
- Check "myTerrain2.xml" for some more informations about the exported tiles (e.g. face count)

- Double click “ShowTerrain.dxstudio” (the meshes are now background-loaded one by one). The loading is finished when object count is 66 (terrain tiles+camera+light)
- Such a big terrain could cause high load on your PC. Later in your application, you would want to load/show/hide terrain tiles based on the viewer’s position, of course.
- Use mouse/WASD to move the camera around. Toggle “Wireframe” and “Normals” for more info (showing the Normals might cause heavy load and slow response)
- Done

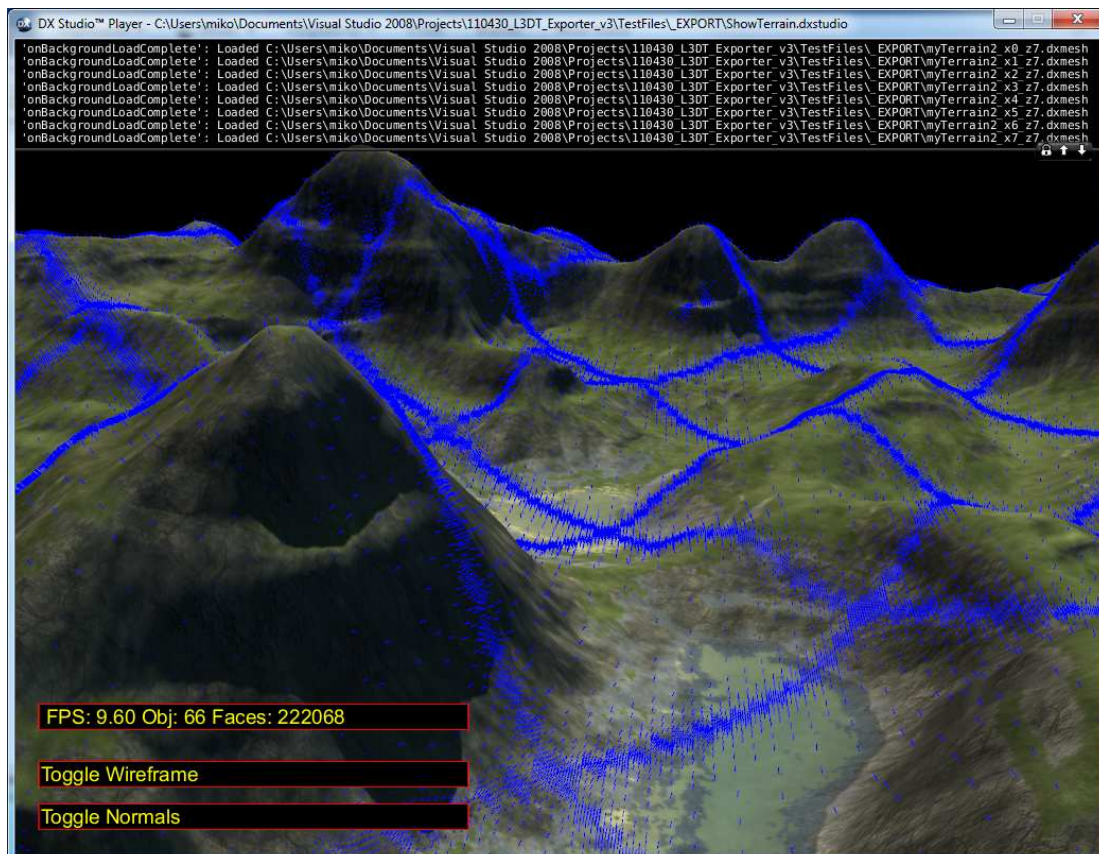
The created terrain in the demo doc



Wireframe mode (see L3DT's face reduction?)



Vertex Normal indicators switched on



Issues/Bugs

- The helper doc “ShowTerrain.dxstudio” does only take first *.xml found. So, it might not work as expected when exporting several projects to the same directory
- Although some measures are taken inside the exporter to hide seams between tiles (e.g. “texture pixel shifting”), these are still notable at close-up view. This might be caused by a “normals problem” at tile edges – or something else. Improvement is needed.
- As DXStudio doesn’t like non-2^n texture sizes, “bridging” between terrain tiles causes a slight shift in Uvs. This shouldn’t be visible, but is not “entirely correct” (in terms of scaling).

Contact/Download

Author: miko@mikoweb.de

Download: <http://www.mikoweb.eu>

L3DT Homepage: <http://www.bundysoft.com/L3DT/>

DXStudio Homepage: <http://www.dxstudio.com/>

Credits

Many thanks to Aaron Torpy, the creator of L3DT, for his support during the development of this exporter. Same goes to the guys over at DXStudio, for helping me getting into the DXMesh API.