Script Version 2.3

Author: Lorne Brooks

LB Tools - Export FBX to Unreal - Guide

This script was created for the purpose of exporting assets to the Unreal Editor and has been used successfully in full games production. It has been tested on both UE3 and UE4.

The main idea of the script is to speed up exporting of static meshes, skeletal meshes, single skeletal animations and batches of separate skeletal animation files. It will attempt to do most of the work for you, with regards to identification and selection of relevant objects within a Max scene. You can however, specify through selection which assets the script should use to export from. This can both speed up the export process and help to control what is saved out.

This is a fairly generic script for exporting and doesn't support advanced features such as source control (perforce), or project specific workflows. However, once you understand how the script works with scene objects your export workflow should improve.

Installation and Setup:

Simply download the script from Scriptspot then drag the MZP file into your Max viewport.

The files will be copied to the following paths with "LBTools_" as the prefix...

C:\Users\<*userName*>\AppData\Local\Autodesk\3dsMax\<*maxVersion*>\ENU\scripts\startup
C:\Users\<*userName*>\AppData\Local\Autodesk\3dsMax\<*maxVersion*>\ENU\scripts
C:\Users\<*userName*>\AppData\Local\Autodesk\3dsMax\<*maxVersion*>\ENU\usericons

The macro in the script file should automatically run and install an interface action without requiring a Max restart. You can find the script interface via the Customize User Interface > Toolbars Tab > Category: **LB Tools**…
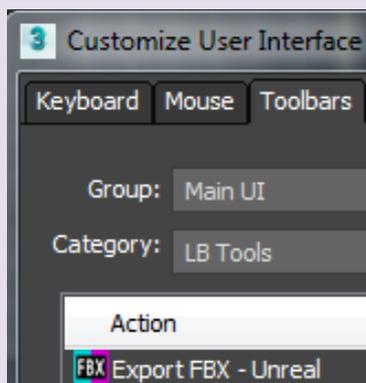
You can then drag this to a toolbar as an icon button…

Key Features:

➢ Define an export path using a variety of methods, or simply export to the scene folder.
➢ Export static meshes and their collisions (supports Convex "UCX_", Box "UBX_", and Spherical "USP_" volume types). *Note:* *It does not actually create the collisions.*
➢ Export skeletal meshes and their bind skeleton (game rig).
➢ Export a single skeletal animation (scene) or batch export from a folder of animations.
➢ Static mesh exports output to a text log file, showing all collisions found and any potential collision name errors.
➢ Skeletal mesh and animation exports report a summary to the Maxscript Listener.
➢ Exporting is non-destructive to the original source – i.e. the original file is not saved if changes are made to the scene for export purposes.
➢ Blendshapes from the Morpher modifier can be exported.
➢ Level of Detail (LoD) utility sets are supported for both static meshes (including their collisions) and skeletal meshes.
➢ Supports up to FBX version 2018. Version 2016 is the default (as per UE4 documentation).

Additional Features:

➢ Specify an export path using a dropdown list of paths stored in a text file. You can specify a root path for the text file dropdown list paths to shorten the full path length.
➢ Specify a limited set of exposed FBX export options via a Settings dialog.
➢ Quickly export using Shift+Click on the menu item or toolbar without opening the interface.
➢ Specify whether static meshes are origin centred, world oriented or pivot centred.
➢ Export static meshes to a single FBX file for convenient storage. Defaults to separate files for each static mesh found.
➢ Export separate meshes as a single attached mesh.
➢ Use selection sets to determine what static meshes to export.
➢ Grouping and linking are supported for static meshes and their collisions.
➢ Export skeletal meshes with either full or partial hierarchy selection.
➢ Export separate skeletal meshes (with a shared rig) as a single export file using a Gestalt method.
➢ Skeletal meshes can be grouped (for convenience) even if not part of a Level of Detail set.
➢ Specify using named selection sets any skeletal meshes that share a skeleton with the Gestalt method. This option allows for multiple hierarchies in the same scene.
➢ Specify what skeletal mesh is used by adding a skin mesh name (both scene and batch exporting are supported).
➢ Specify an animation name for output to override using the maxfile name.
➢ Use time tags to export a combined animation made up of multiple sequences as separate sequences based on the tag name and range specified.
➢ Batch export skeletal animation from a selection of maxfiles chosen from a list and sorted by either name or date. You can recurse all subfolders. Also supports time tags.
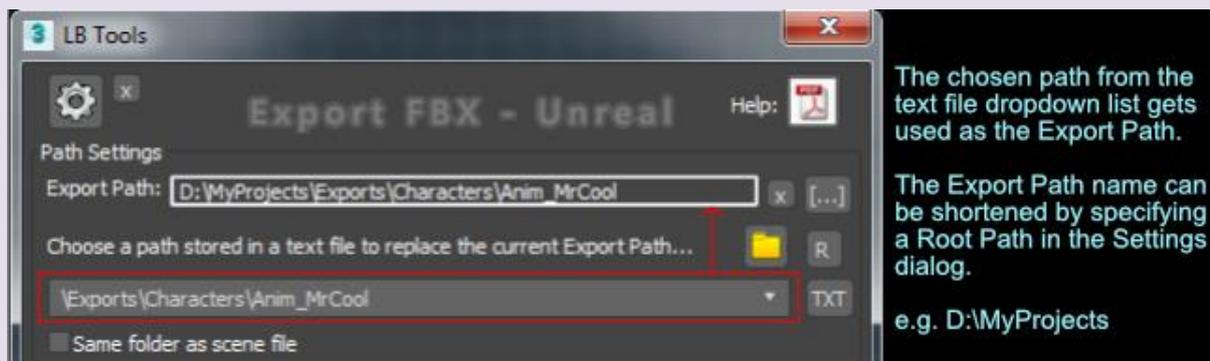
Some Script Limitations:

➢ Does not do anything with materials.
➢ Does not support source control (e.g. Perforce).
➢ Does not deal at all with the Unreal Editor or game engine. It only handles content exporting from 3ds Max.
➢ Does not support skeletal animation exporting using multiple takes in one FBX file.
➢ Does not support exporting of sockets.

Interface:
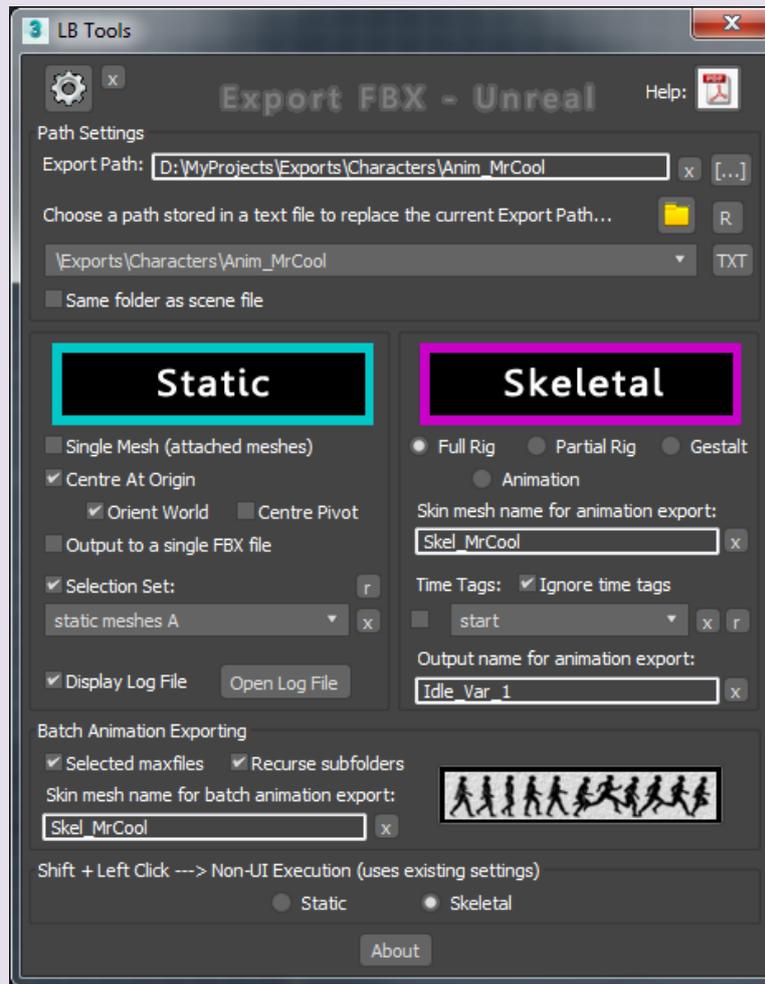
Path Settings:

An edit box called **Export Path** allows you to directly copy/paste a folder path from Explorer into it. <u>The path must not have a backslash at the end</u>. Alternatively, you can browse for a path using the **[…]** button next to the edit field. The **"x"** button will clear the edit field text.

Below that is a method of storing a list of favourite paths which are accessed from a text file stored in **"C:\Temp\LBTools_ExportPaths.txt"**. The text file can be quickly edited using the **"TXT"** button.

A dropdown list is populated from the stored text file. The list of paths in the text file must not have backslashes at the end. The list displayed for each entry is automatically shortened by removing the Root Path substring of the full path string. The Root Path is set using the Settings button (gear icon) found at the top left of the interface which opens a new dialog.

An example full Export Path would be…

> D:\MyProjects\Exports\Characters\Anim_MrCool {no backslash}

…where the Root Path substring is… D:\MyProjects {no backslash}

If you make changes to the text file while the interface is still open, you can quickly update the list by pressing the **"R"** button. Otherwise you can just reload the UI.

Note: Export folders are created manually – the script will not create them for you.

The **"Same folder as scene file"** checkbox will output the FBX files to where the currently open scene is saved. This option overrides the Export Path. A warning message is displayed when this option is active, as you may forget you left it on otherwise. By default, the Export Path settings are the preferred method to be used.

Export Buttons:

Simply press the relevant button (**STATIC** or **SKELETAL**) when ready to export. The file(s) will be output according to either the Export Path or the Same folder as scene file (if that option is checked).

Alternatively, you can use *Shift + Left Click* on the toolbar icon when the interface is closed to execute a quick export. It uses the last saved settings from the interface.

Choose which method will respond to the quick execution at the bottom of the UI...
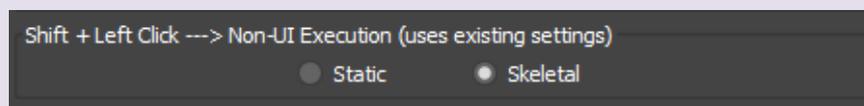

Things to note generally:

➢ Some of the functionality is contextual to the interface options or whether you have objects selected. Understanding this behaviour is important to using the script successfully.

➢ If any of the export files are read only, they will be made writable and overwritten without prompting. A summary will be presented at the end of exporting telling you if new files were added, existing files overwritten or overwritten files which also need to be reconciled (i.e. they were read only in Perforce for example).

Example summary: In this case the Box.fbx has been overwritten, but requires reconciliation (checking out) in Perforce (or made writable) because it was not checked out for editing first...
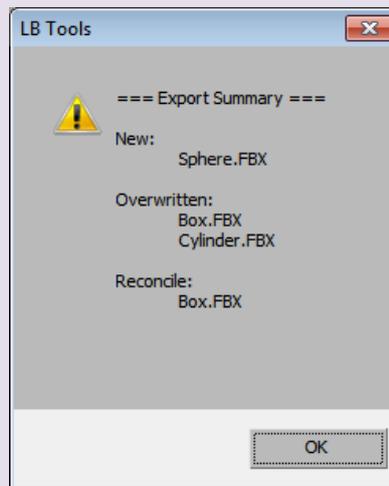
Export Settings:

These options affect either the **STATIC** (left) or **SKELETAL** (right) export buttons.
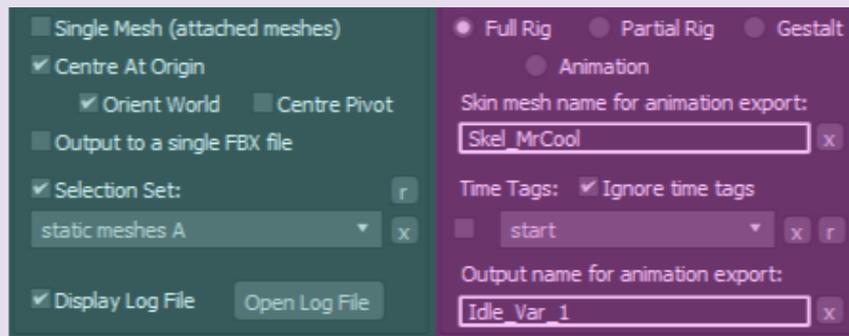
STATIC export settings:

When exporting using the Static option, you can either select what you want to export or just let the script find any valid geometry. The script will attempt to locate and select any validly named collision volumes* associated with each static mesh or Level of Detail* set. It will report the output results to a log file for your analysis.

- **Single Mesh** – Creates duplicates of the current selection, attaches them into a single mesh, exports the single mesh to a single FBX file based on the maxfile name then deletes the new mesh leaving your original scene intact.

- **Centre At Origin** – This will reposition static meshes to the origin and either align their pivots to the centre of the mesh or leave the pivots as is. This is dependent on the sub-options for either **"Orient World"** or **"Centre Pivot"**. Especially useful if you have scenes setup so that objects are offset from the origin – for ease of viewing. Also affects the Single Mesh option. *Note: This option also moves any associated collision volumes by first linking them to their static meshes. There's no need to manually link them prior to exporting.*

- **Output to a single FBX file** – If checked, all static meshes and collisions found will be saved into a single FBX file. The output name will be the maxfile name with the added prefix 'StaticMeshes_'. For example: *Building_A.max = StaticMeshes_Building_A.FBX*

- **Selection Set** – Uses collections of static meshes created with selection sets accessible via a drop-down list. A chosen selection set will be used to select the static meshes to export as though selecting them manually. This allows meshes to be isolated from other scene geometry, frozen or hidden without requiring manual selection. This option overrides the default behaviour of using either manual selection or exporting all the scene geometry as static meshes. The **"r"** button will refresh the list in case the scene data has changed. The **"x"** button will delete all scene selection sets and clear the list.

- **Display Log File** – Displays a text file after exporting static meshes. It is stored in **"C:\Temp\LBTools_ExportLog.txt"** and contains information on what static meshes were exported and their associated collision volumes. There's an **"Open Log File"** button which will open this text file for quick editing. This excludes Single Mesh exports.

SKELETAL export settings:

- **Full Rig or Partial Rig** – When exporting skeletal meshes, the output will usually be a single FBX with the skin mesh name as the saved filename. The scene must have a valid skin mesh (using the Skin modifier) containing assigned bones. Depending on user scene selection, it will determine the appropriate skin mesh and skeletal hierarchy to export. If nothing is selected, it will search the scene for valid skin meshes and operate on those. The script also handles exporting multiple skinned meshes at once and will output these as separate files using the skin mesh name for each as the filenames. The mode chosen (Full Rig or Partial Rig) will determine whether to include the entire hierarchy of bones or use only the skin bones found in the skin mesh.

  Note: When exporting using Partial Rig mode, FBX will always include the parent bones up to the root even if exported manually without this script.

- **Gestalt –** Allows exporting of separately skinned meshes that share a hierarchy as a single skeletal mesh. You can have multiple hierarchies in the scene in this case.

  Three methods for export using the Gestalt option include:

  ➢ Selection Sets; Finds all user defined selection sets tagged using the prefix "FBX_" followed by the output name for collections of skinned meshes that share the same hierarchy. Nothing should be selected initially for this option to work.

  ➢ Nothing selected; Finds all hierarchies and exports a single FBX file containing a single skeletal mesh for each hierarchy which in turn are made up of separately skinned meshes. Uses the name of the maxfile as output.

  ➢ Selected objects; Finds any skinned meshes in the selection and if they share the same hierarchy, exports them as one mesh. Uses the name of the maxfile as output.

- **Animation** – Exports bone animations using the name of the maxfile as the output for the FBX animation file. There's an edit field provided to allow the animator to specify a valid <u>skin mesh name</u> to export from should there be more than one skin mesh in the scene. This saves the animator from needing to select a skin mesh in case there are multiple skinned meshes skinned to different rigs. For example; choosing either a weapon rig or a character body rig. If no skin mesh name is provided and nothing is selected when exporting a multi-hierarchy scene, a dialog will appear prompting you to select a skin mesh name to use – along with a continue process button.

    You can also specify an <u>output name</u> to override the maxfile name default behaviour. This can be useful in cases where you have multiple animations in one scene (like cinematic work) and you don't want to use the maxfile name as the sequence. The **"x"** buttons clear the edit fields and focus on them for quick editing.
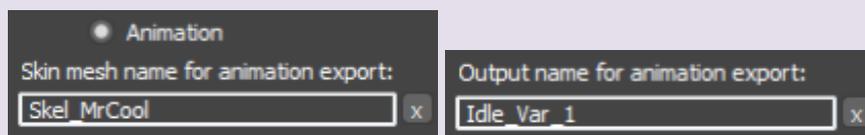
- **Animation: using scene Time Tags –**

Correctly formatted time tags are supported for animation exporting of composite sequences. For example, you may have a single maxfile which contains animation for a start, loop and end sequence on different time frames. You can have any number of sequences and their times can even overlap.

This process can make animation editing easier as you only need to edit one maxfile. The Animation mode will automatically look for any time tags and export each sequence using the tag name and the range specified within the tag. The **"Ignore time tags"** option can be used with both Animation mode and Batch Animation – it simply exports a single FBX file using the maxfile name, irrespective of any existing time tags, in case tags were used for time navigation instead.
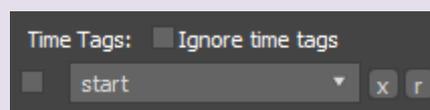

The time tags must conform to a specific format to be recognized for this purpose. The format should be in the form… tagName[space]startTime[dash]endTime. e.g. Jump_Start 0-50

When setting up <u>Time Tags</u>:

  ➢ No spaces should be in the name, use underscores instead.

  ➢ Only one space should exist between the name and the start time.

  ➢ Use a dash to separate the start and end times.

  ➢ Time ranges can overlap and don't need to relate to one another.

Another feature using time tags is the ability to only export a specific sequence by selecting a tag name in a dropdown list (below the skin mesh name field). There's a checkbox next to the list to activate this feature. If unchecked, all time tag sequences will be exported. The **"r"** button will refresh the list in case the scene data has changed. The UI will also populate the list when first opened, if any tags are found. The **"x"** button deletes any existing scene time tags.
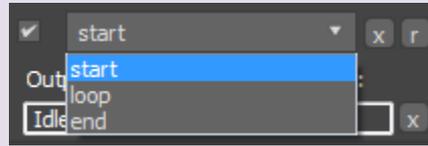
Note: These two settings don't get stored in the interface INI file when changed. However, you can invoke these to export a single sequence using the **Shift + Left Click** method to execute an export, but if you restart the interface the settings are reset.

Important Note: *For exporting to use the correct time ranges as specified by either time tags or the scene range, keys should not exist outside of the scene range or they will be included in the FBX export. This is just how Autodesk FBX works by default. To get around this, it's best to skin the mesh to a skeleton that has no keys on it and use a control rig to key with. The skin rig will be constrained to the control rig. The script will only select the skin mesh and the keyless skeleton driven by the control rig which will respect the scene range or time tags.*

Batch Animation Exporting:
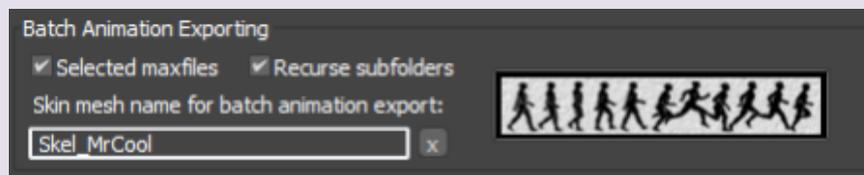
Batch Animation Exporting button…

This section is for animators to be able to quickly export from a folder of export ready* skeletal animation maxfiles. It assumes the export rig that the mesh is skinned to is in the scene and the skin is valid (has bones assigned).

There's an edit field to allow the animator to specify a skin mesh name if there are multiple skinned meshes in the scene. Only the specified mesh (if valid) will have its animation exported. Leave it blank if you know there's only one skinned mesh in the animation scenes.

Note: This process is not as comprehensive at selecting a valid skin mesh when compared to exporting a single animation file using the Skeletal button. It will simply attempt to find the first skin mesh to export from. If multiple skin meshes on different rigs are in the scene you should specify a skin mesh name to reference the export rig from.

The animator can also enable a **"Selected maxfiles"** option, so when the batch export process is run a new dialog (shown below) will display listing all the maxfiles in the chosen source path. Simply select names from the list then hit the export button below it. If this option is off all maxfiles found in the chosen source path will be exported and the dialog won't appear.

The list can be sorted by either name or last date modified. There is also a browse button **[...]** for reassigning the source path. File output will either be to the specified Export Path or the current scene folder if the "Same folder as scene file" option is checked.

If the **"Recurse subfolders"** option in the main UI is checked, the script will traverse any subfolders within the chosen source path and export all maxfiles found. It will automatically create new subfolders in the Export Path to match the same subfolder structure as the source path. If the selected maxfiles option is also used, the dialog displays the subfolders as a prefix next to the maxfile names, unless they are in the root of source path. The recurse subfolders option can be toggled while the selected maxfiles dialog is displayed and will update the list.

Example dialog for Batch Animation Exporting using the selected maxfiles option…
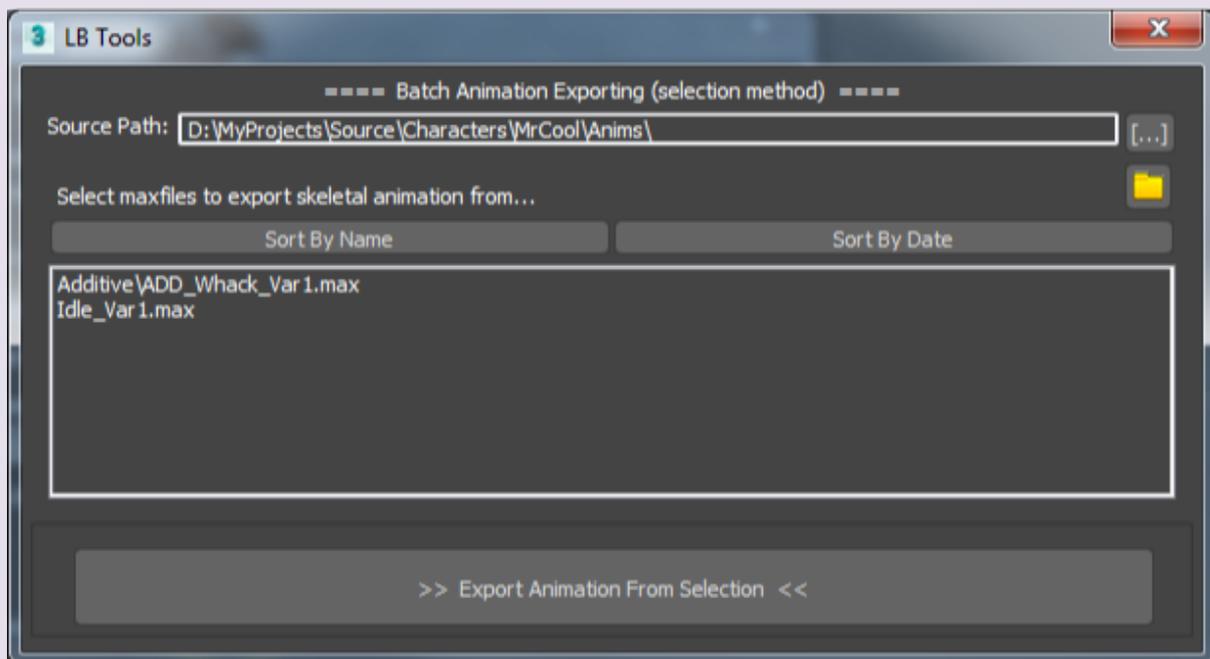
Paths Setup…

Source Path… D:\MyProjects\Source\Characters\MrCool\Anims\

Export Path… D:\MyProjects\Exports\Characters\MrCool\Anims\

Dialog display…

Idle_Var1.max *{this file exists in the source "root" path}*

Additive\ADD_Whack_Var1.max *{this file exists in a subfolder Additive under the source "root" path}*

Script Version 2.3

<u>Output would be…</u>

D:\MyProjects\Exports\Characters\Anim_MrCool\Additive\ADD_Whack_Var1.FBX

D:\MyProjects\Exports\Characters\Anim_MrCool\Idle_Var1.FBX

The subfolder Additive would be automatically created if it doesn't already exist.

Maxscript Listener Output:

Using the Skeletal Mesh, Gestalt Mesh, Animate (with or without time tags), Batch Animation or Single Mesh (Static) options will also result in an output to the maxscript listener dialog. Static meshes with collisions will use the log file for information output.

Settings Dialog:
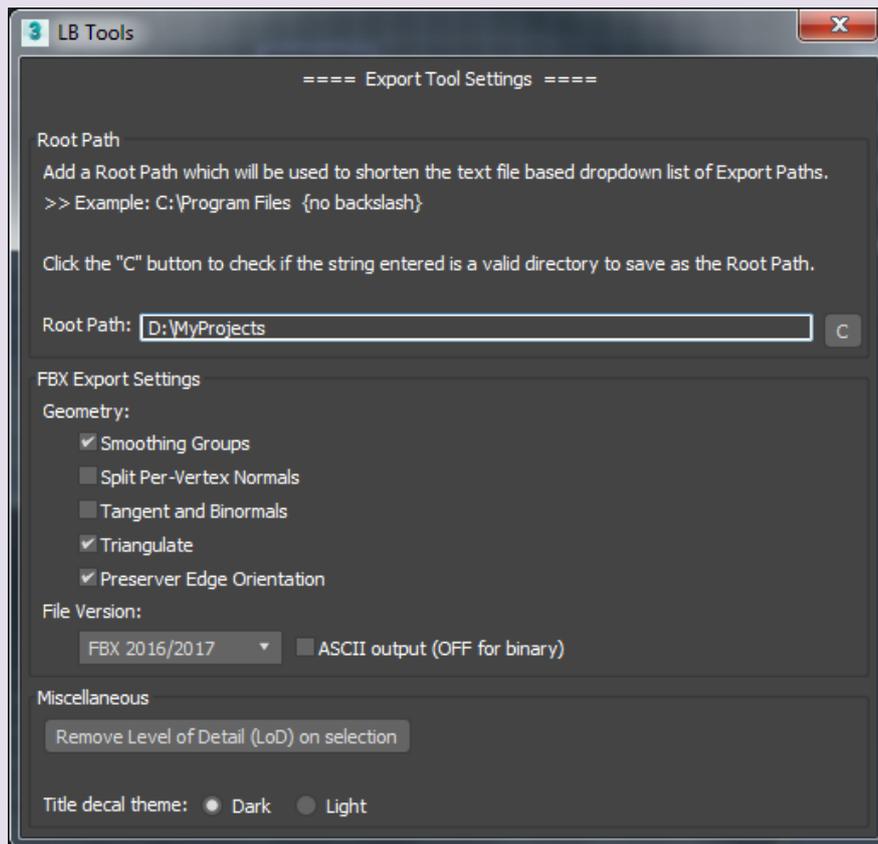
A **"Settings"** button (gear icon) can be found at the top left of the main interface which will open another dialog.

It contains the <u>Root Path</u> definition used by the text file display method (described above) as well as exposed FBX Export Settings. There's also a title decal theme to option depending on your workspace colour setup (requires a restart of the UI).

- **FBX Export Settings –**

These include some exposed geometry settings, used for both static and skeletal mesh exporting. You can also define the output file version of the FBX and whether to use ASCII format or leave as binary.

- **Remove Level of Detail (LoD) on selection –**

Removes association with the Level of Detail utility plugin for any selected LoD groups. The groups will be exploded and all LOD meshes will be made visible.

---

*Naming Static Mesh Collision Volumes:

Collision volumes for static meshes should adhere to the Unreal FBX Static Mesh Pipeline for naming conventions. The script will attempt to parse the name of the collisions and if found valid will export it associated with its static mesh based on the name. Any potentially invalid collisions are reported to the log file.

The script will only accept naming conventions that Unreal itself would accept (on import), had you exported your static mesh and collisions manually through the FBX interface.

Here's an example log file output from a static mesh with valid and invalid collisions…

For more information on naming, see the section on Collision > Caveats and Considerations > RenderMeshName found on the Unreal website…

https://docs.unrealengine.com/latest/INT/Engine/Content/FBX/StaticMeshes/index.html

*Level of Detail (LoD):

A LoD is a special group of objects created using the Level of Detail utility plugin. The user first defines a group of LOD objects (of varying mesh densities), then creates a set using the LoD utility with defined distances. Unreal supports the auto-importing of LOD's into both static meshes and skeletal meshes, if exported as part of a single FBX. The script will support both of these cases.

For more details see…

[How To Import Static Mesh LODs](#)

[How To Import Skeletal Mesh LODs](#)

[Creating and Using LODs](#)

In order to properly export static mesh LoD groups along with their collisions using the script, please be aware of the following…

➢ Collision volumes should be named according to the LOD names within the group, not the group name. Otherwise, normal naming conventions apply as per non-LoD static meshes. Unreal will not recognize collision meshes if they are not named correctly.

➢ If there is more than one LoD object in the scene that has collision meshes, those collision meshes must be linked to their associated LoD. This is so that the script can identify which collision meshes belong to which LoD object, in case their LOD's share the same names as other LoD set LOD meshes.

➢ A single LoD with collisions in the scene does not need to have its collisions linked.

➢ Note that having anything linked to the LoD that is not part of the set will cause Max to crash. However, as stated above for multiple LoD's in the scene the script will handle this, but only for valid collisions.