

# ***TEXTMESH PRO!***

Advanced Text Rendering for Unity 3D

## **USER GUIDE**

**Release 1.0.54**

*Dec 2016*

*Copyright (C) 2016 Stephan Bouchard - All Rights Reserved*

This manual, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. The content of this manual is furnished for informational use only, is subject to change without notice and should not be construed as a commitment by its authors. The author assumes no responsibility or liability for any errors or inaccuracies that may appear in this manual.

# TEXTMESH PRO!

## User Guide / Table of Contents

### Getting Started

Overview .....	4
Installation .....	4
Quick Start.....	4
Support & API documentation .....	4

### TextMeshPro Component

Text Input Box .....	5
Font Settings .....	6

### Material Editor & Shaders Overview

Shader Overview .....	7
Material Editor UI Panels & Features .....	7

### Font Asset Creator

General Overview .....	10
------------------------	----

### Font Asset

General Font Information & Sub-Assets .....	13
Font Settings .....	13
Character Set Info.....	14
Kerning Table .....	14

# TEXTMESH PRO!

## User Guide / Getting Started

### Overview

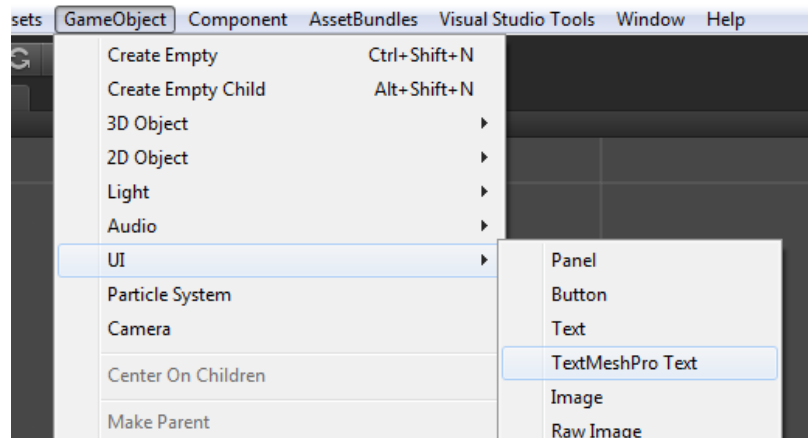
This User Guide was designed to provide TextMesh Pro users with a basic overview of the features and functionality of the tool.

### Installation

Once you have downloaded TextMesh Pro from Unity's Asset Store, go to: **"Assets->Import Package->Custom Package..."**. In the Import Asset window, find and select the TextMeshPro.unitypackage file. After the "Importing package" window appears in Unity, verify that all items to import are selected and then click the Import button in the bottom right of the window.

### Quick Start

There are two TextMesh Pro components available. To add a new TextMesh Pro text object which works with the Mesh Renderer, go to: **"GameObject->3D Object->TextMeshPro Text"**. To add a TextMesh Pro Component which works with the New UI, go to: **"GameObject->UI->TextMeshPro Text"**.



You may also wish to watch this [short video](#) showing how to add TextMesh Pro text objects into a scene.

### Support & API Documentation

Should you have questions or require assistance, please visit the [TextMesh Pro User Forum](#) where you will find additional information, [Video Tutorials](#) and [FAQ](#). In the event you are unable to find the information you seek, feel free to register and post on the user forum or to contact [Support@DigitalNativeStudios.com](mailto:Support@DigitalNativeStudios.com)

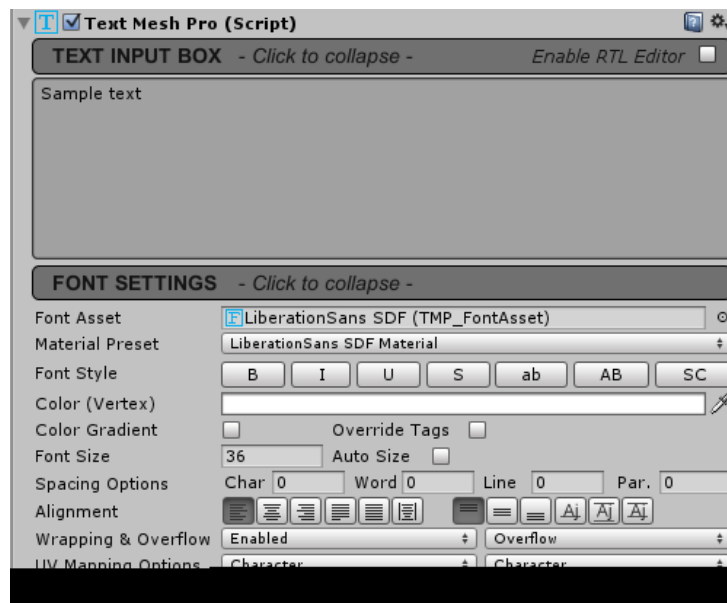
[Online Documentation](#) is also available on TextMesh Pro including Rich Text tags, Shaders, Scripting API and more.

# TEXTMESH PRO!

## User Guide / Component Overview

### Component Overview

The TextMesh Pro component is divided into two sections: The Text Input and the Font Settings.



### Text Input Box

This area is where you will type the text to be rendered. Multiple lines of text can be entered by pressing <Enter> on your keyboard or by typing the special character \n.

Tabulation can be achieved by using \t. These are tab stops with a spacing interval equal to roughly 25% of the point size. For an alternative to tabs, see the <pos> tag below.

The Text Input area also supports [Rich Text tags](#). Tags are defined by enclosing the tag between "<" and ">" characters. Below is a partial list of supported tags. Updated tag list is available [here](#).

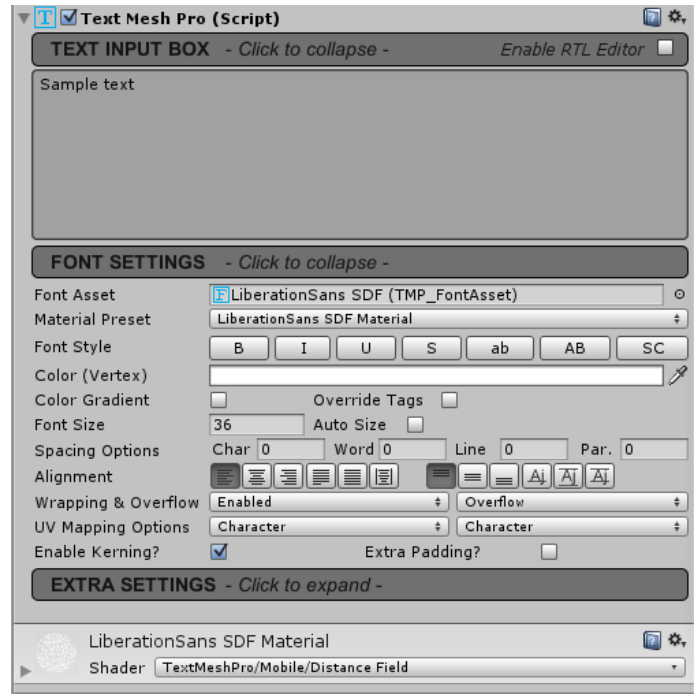
<b>Bold</b>  
<i>Italics</i>  
<u>Underline</u>  
<s>Strikethrough</s>  
<sup>Superscript</sup>  
<sub>Subscript</sub>  
<size=48>Point size 48</size>  
<size=+18>Point size increased by 18</size>  
<size=-18>Point size decreased by 18</size>  
<pos=5.5>Characters positioned about 5.5 characters from left side.  
<color=yellow>Yellow text</color>  
<#00ff00>Green text</color>

# TEXTMESH PRO!

## User Guide / Component Overview

### Font Settings

<b>Font Asset</b>	This is the TextMeshPro Font Asset that will be used to render the text.
<b>Font Style</b>	Applies a global style to the text like bold, italics, etc.
<b>Vertex Color</b>	Color that will be assigned as a vertex color for each character unless a color tag has been specified.
<b>Color Gradient</b>	Enables using a Vertex Color Gradient.
<b>Override Color Tags</b>	Enables overriding color tags. This results in all characters using the Face Color instead of the color tag.
<b>Font Size</b>	The point size of the characters.
<b>Auto Size</b>	Dynamically adjusts the point size between the min & max to fit the size of the text container.
<b>Character Spacing</b>	Controls the spacing between individual characters.
<b>Spacing</b>	Enables adjustments to character, line and paragraph spacing.
<b>Alignment</b>	Controls the alignment of the text. Options are Left, Center, Right and Justified. When Justified is selected, the ratio controls how additional spacing is added between words and characters.
<b>Enable Word Wrap</b>	Enables or disables Word Wrapping.
<b>UV Mapping Options</b>	Controls how the texture (face & border) are mapping on the text object.
<b>Enable Kerning</b>	Enables or disables Kerning if any kerning pairs are defined in the kerning table of the Font Asset.
<b>Extra Padding</b>	Extra padding may be required for very small fonts. This extra padding prevents the characters from being clipped.



# TEXTMESH PRO!

## User Guide / Material Editor & Shader Overview

### Material Editor & Shaders

TextMeshPro includes several custom shaders along with a custom material editor which was designed to keep things simple by only exposing those properties which are expected to be modified.

There are two groups of shaders which include a Bitmap shader & a few Signed Distance Field shaders. TextMesh Pro's shaders can be found in "**TextMeshPro/...**" section of the shader list in the material component. Below is a list and brief description of the SDF shaders. Updated shader information is also available [here](#).

#### Distance Field Mobile

This lightweight shader contains just the bare essential features and offers the best overall performance.

#### Distance Field

This more advanced shader includes a large selection of features. It is the shader that is assigned by default to all Signed Distance Field font assets. It still offers excellent performance.

#### Distance Field Surface

This shader includes essentially the same features as the Distance Field shader. However, instead of using simulated lighting, it uses the scene lights. These text objects can also cast shadows. This shader uses per pixel lighting.

#### Distance Field Surface Mobile

This is a lightweight version of the Surface shader with limited features. It uses per vertex lighting and does not use directional information from lights. However it does factor in light distance.

### Material Editor Feature Panels

In this section, you will find an overview of each of the feature panels used by the custom material editor. Clicking on these panels will hide or reveal their features. Some of these panels need to be enabled to turn on their functionality. Different panels will be available on the various shaders.

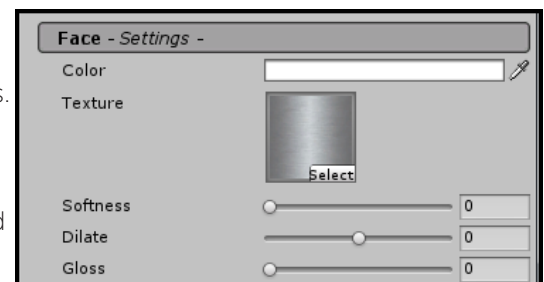
#### Face Panel

**Color** - Select the color and transparency for the face of the characters. This color will be combined with the vertex color.

**Texture** - Select a texture to be applied to the face of the characters. The Color will affect the color of the texture. How the texture is applied to the characters / text object is controlled by the mapping options in the Editor Panel. The texture option is not available on all shaders.

**Softness** - Controls the softness of the face of the characters. Please not the face softness begins at the edge which may be underneath the Outline.

**Dilate** - Increases or decreases the size of the characters.



# TEXTMESH PRO!

## User Guide / Material Editor & Shader Overview

**Gloss** - Controls how shiny the surface of the characters appears when using the Surface shaders.

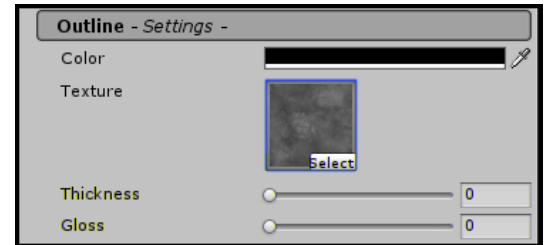
### Outline Panel

**Color** - Controls the color and transparency for the Outline of the characters.

**Texture** - Select a texture to be applied to the face of the characters. The Color will affect the color of the texture. How the texture is applied to the characters / text object is controlled by the mapping options in the Editor Panel. The texture option is not available on all shaders.

**Thickness** - Controls the thickness of the Outline. The Outline is applied on the edge of the face and expands inwards and outwards.

**Gloss** - Controls how shiny the surface of the characters appears when using the Surface shaders.



### Underlay Panel

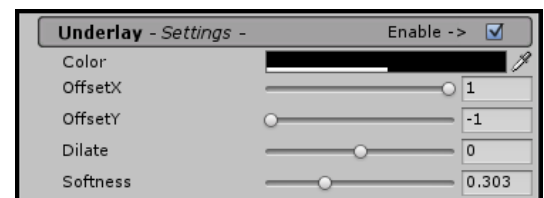
Underlay can be used to add a shadow or border to the text object. This is a very efficient way to add contrast to small text.

**Color** - Controls the color and transparency for the Underlay.

**Offset (X, Y)** - Controls the position of the Underlay. Please note Underlay has a limited range.

**Dilate** - Increases or decreases the size of the underlay. This can be used to make the shadow appear bigger or with an Offset of (0, 0) create a nice border around the text.

**Softness** - Controls the softness of the underlay.



### Bevel Panel

Bevel simulates the visual appearance of a 3D bevel on a 2D object.

**Type** - Two types are available; Outer and Inner bevel.

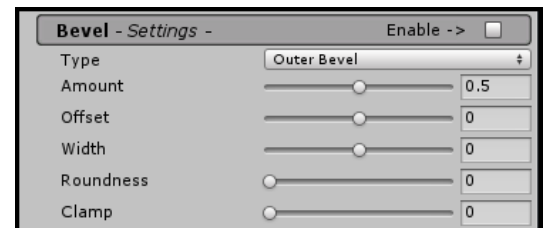
**Amount** - Controls the amount of bevel being applied.

**Offset** - Controls the position of the bevel relative to the edge of the face of the character.

**Width** - Controls the width of the bevel effect.

**Roundness** - Determines if the bevel ridges will appear sharp or rounded.

**Clamp** - Limits the height of the bevel. This visually looks like the edges have been flattened.





# TEXTMESH PRO!

## User Guide / Material Editor & Shader Overview

### Lighting Panel

The lighting settings controls the appearance of the bevel, bump and environmental mapping.

**Light Angle** - Controls the angle of the light.

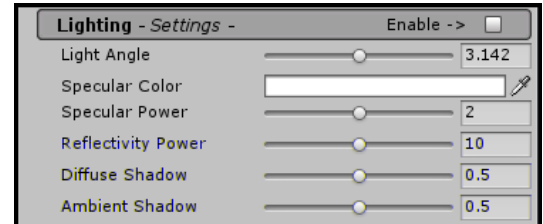
**Specular Color** - Determines the color of the specular highlights.

**Specular Power** - Controls the strength of the specular highlights.

**Reflectivity Power** - Controls the amount of light being reflected.

**Diffuse Shadow** - Increase or decrease the amount of light received by the object.

**Ambient Shadow** - Controls how pronounced the light and bevel edges appear.

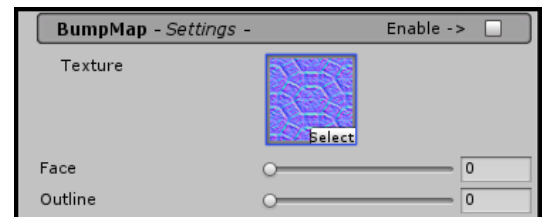


### BumpMap Panel

**Texture** - The normal map to be applied to the text object.

**Face** - Controls the amount of bump mapping to be applied to the face of the characters.

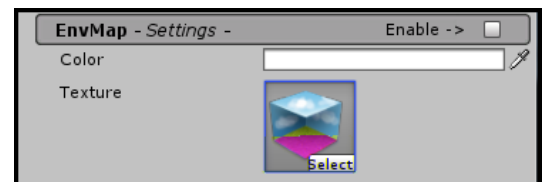
**Outline** - The amount of bump mapping applied to the Outline of the characters.



### EnvMap Panel

**Color** - The color will brighten, darken or tint the environmental map. Default color is black which leaves the text object unaffected.

**Texture** - The Cube map to be applied to the text object.



### Glow Panel

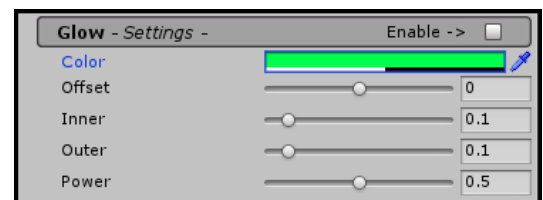
**Color** - The color and transparency of the Glow.

**Offset** - Controls the position of the glow. A value of 0.0 is on the edge of the face of the character.

**Inner** - Controls how far the Glow effect extends towards the inside of the face.

**Outer** - Controls how far the Glow effect extends towards the outside of the face.

**Power** - Controls the amount of Glow.

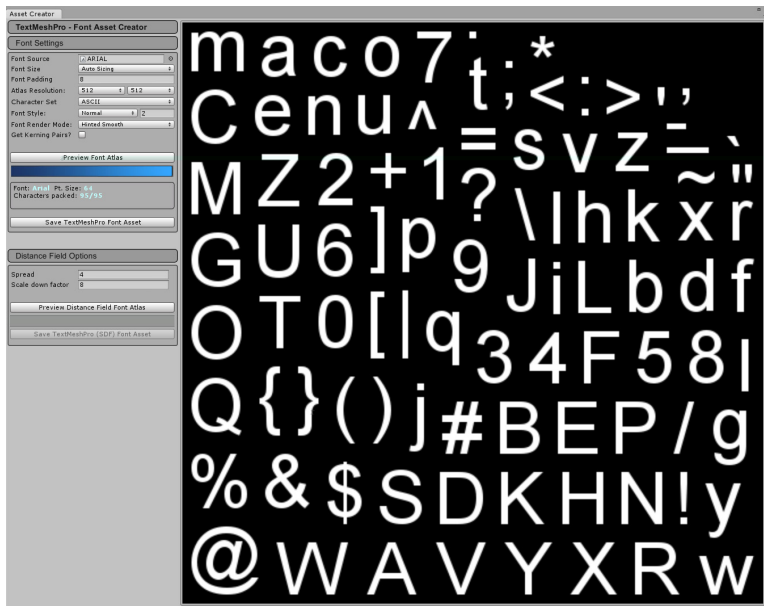


# TEXTMESH PRO!

## User Guide / Font Asset Creator

### Font Asset Creator Overview

To access TextMesh Pro's Font Asset Creator go to: "Window->TextMeshPro Font Asset Creator". The following video provides a detailed overview of the [Font Asset Creation Process](#).



Online documentation is also available on this topic at this [location](#).

It is also strongly recommended that you view the following video about [Creating & Working with Material Presets](#) with TextMesh Pro.

### Font Settings

The Font Settings panel is where the font for which a font asset is to be created is selected along with various settings.

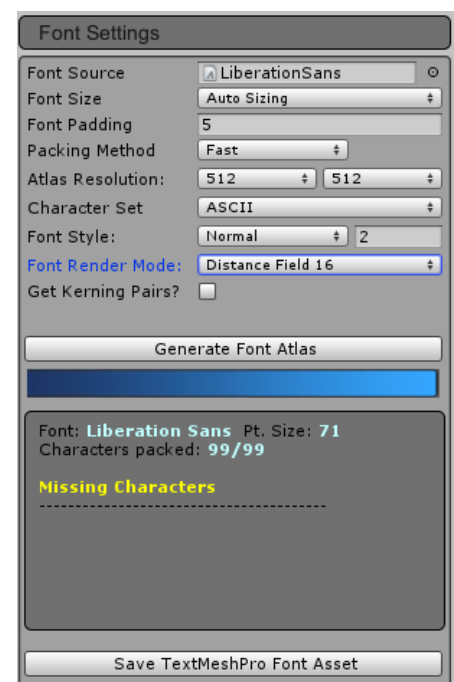
**Font Source** is where the font for which an asset is to be created is selected.

**Point Size** determines at what point size the font will be created. You can either manually select the size of the font or use the Auto Sizing mode which will find the largest possible font size to fit in the given Atlas Resolution. It is recommended to use Auto Sizing.

**Font Padding** determines how much space in pixel units between each character. Font Padding also determines the spread used in the Signed Distance Field Font Render Mode. For an Atlas Resolution of 512 X 512, a value of 5 for padding is generally adequate.

**Atlas Resolution** determines the size of the Font Atlas. For most fonts which include the full ASCII character set, a size of 512 X 512 works usually well for Signed Distance Field Render Mode.

**Character Set** determines what characters will be included in the font atlas.



# TEXTMESH PRO!

## User Guide / Font Asset Creator

**Custom Range** allows you to define a range that represents which characters to be included. To define a range the includes the ASCII set + Latin supplemental characters, a user would enter 32-126, 161-255 in the Custom Range window.

**Custom Characters** allows the user to specify characters to include by typing them Custom Character window.

**Characters from File** enables a user to supply a plain text file that contains all the characters to be included in the atlas.

**Font Style** allows the user to select if the characters will be created as bold, italics or as an outline as well as defining the thickness of the outline or boldness. Keep in mind that this is mostly useful for normal bitmap fonts. Since Signed Distance Field font rendering enables the creation of bold or an outline dynamically these style options are fun to experiment with but not essential.

**Font Render Mode** determines how the characters will be rendered.

**Hinted Smooth** is anti-aliased rendering with hinting.

**Smooth** is the same without hinting.

**Raster Hinted** renders the characters without anti-aliasing with hinting.

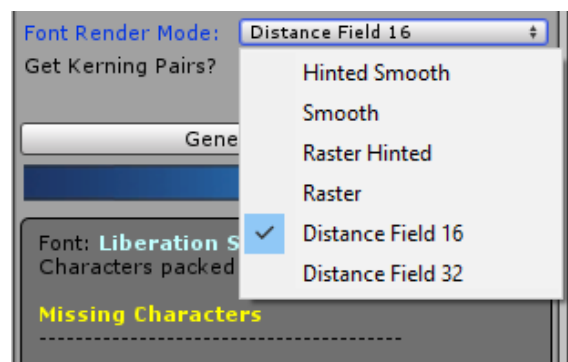
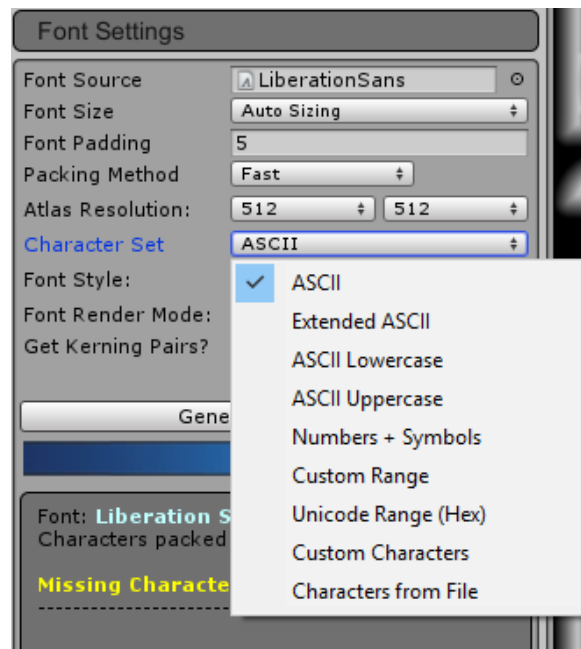
**Raster** is no anti-aliasing or hinting.

**Signed Distance Field 16** is the standard mode for creating a SDF Atlas. The font will be up sampled 16X to produce a more accurate Signed Distance Field Atlas.

**Signed Distance Field 32** is the same as above but with 32X up sampling. Using this mode will result in a more accurate SDF Atlas which although more accurate may deliver only subtle visual improvements. This options should mostly be used when trying to fit a full ASCII set into a small 256 X 256 atlas or when dealing with a complex shaped font with high frequency curve changes. Note: This processing mode will be significantly slower than the 16X mode.

**Note:** Padding represents the spread that will be used in the creation of the SDF Atlas. The larger the spread the longer it will take to process the atlas. Higher spread provides for better control over outline thickness, bevel and glow "tweak-ability" but will result in the characters being smaller in the atlas. Smaller point size means potentially less sampling points and perhaps not as good contour accuracy. It is a trade off or balance between those two. Typically a Spread or padding of 5 is good for normal text and Spread of 7 for things like titles which typically have larger outline, bevel and glow options.

**Get Kerning Pairs** determines if the kerning pairs that may exist in the font file will be added to the font asset. In the event no kerning pairs are found in the font file, new kerning pairs can always be added in the TextMeshPro Font Asset Editor Panel.



# TEXTMESH PRO!

## User Guide / Font Asset Creator

**Generate Font Atlas** will get the font engine to render the select font / characters.

**Save TextMeshPro Font Asset** will bring up a file requester allowing you to choose a file location and a name for the newly created asset. By default the font name will be that of the font name. When using Signed Distance Field Render Mode, this name will be Font Name with SDF added. We recommend you save your font assets in the **"Assets/Resources/Fonts/..."** or create such folder structure if one does not exists. As text objects are frequently created via scripts which results in font assets not being referenced by any scene objects, this guarantees your fonts assets will be include in the build and available.

# TEXTMESH PRO!

## User Guide / Font Asset

### Font Asset

TextMesh Pro uses a custom font asset which contains information necessary to display the font. The font asset is divided into five sections which are as follows:

**Face Info** contains information specific to the font that was rendered using the Font Asset Creator. This includes the follow information:

**Font Source** - The name of the font.

**Font Size** - The point size at which the font was rendered in the font atlas.

**Line Height** - This is the spacing between each lines of text.

**Baseline** - The position where the characters "sit" on the line.

**Ascender** - The highest point of any characters in the font.

**Descender** - The lowest point of any characters in the font.

**Underline** - The position of the underline.

**Width & Height** - The width and height of the font atlas.

**Font Sub-Assets** displays the sub assets that are part of the custom font.

**Font Atlas** - Texture which contains all the characters of the font. This texture is specific to such font.

**Font Material** - Default material for this font which holds a reference to the Font Atlas texture.

**Face Style** contains style properties for the font.

**Normal & Bold weight** - This defines the normal & bold weight of the font.

**Italic Style** - Defines the slant of italic.

The screenshot shows the 'TextMesh Pro! Font Asset' editor window. It is divided into several sections: 'Face Info' with fields for Font Source (Liberation Sans), Point Size (86), Scale (1), Line Height (98.90625), Ascender (77.84375), Cap Height (59.1875), Baseline (0), Descender (-18.21875), Underline Offset (-12.26172), Strikethrough Offset (23.675), Superscript Offset (77.84375), Subscript Offset (-12.26172), Super / Subscript Size (0.5), Padding (9), Width (1024), and Height (1024). 'Font Sub-Assets' shows Font Atlas as 'LiberationSans SDF Atlas' and Font Material as 'LiberationSans SDF Material'. 'Font Weights' is collapsed. 'Fallback Font Assets' is also collapsed. 'Settings used to simulate a typeface when no font asset is available' shows Normal Weight (0), Spacing Offset (0), Italic Style (35), Bold Weight (0.75), Bold Spacing (7), and Tab Multiple (10). 'Glyph Info' and 'Kerning Table Info' are also collapsed.

# TEXTMESH PRO!

## User Guide / Font Asset

**Glyph Info** contains specific information about each character in the font.

**ASC II & Char** - The ASC II code and Character.

**X & Y** - The X and Y position of the character in the font atlas.

**W & H** - The Width and Height of the character in the font atlas.

**Offsets** - The X Offset is where the character is drawn relative to (0, 0). The Y offset represents where the top of the character is located relative to the baseline or (0, 0). The Advance is how far to advance after this character.

**Glyph Info** (Click to collapse)

Ascii: 32	Char: [ ]		
X: 8	Y: 519	W: 0	H: 0
Offset Values:	X: 0	Y: 0	Ad: 18.34375

Ascii: 33	Char: [ ! ]		
X: 84	Y: 379	W: 7.15625	H: 47.25
Offset Values:	X: 5.65625	Y: 47.25	Ad: 18.34375

Ascii: 34	Char: [ " ]		
X: 238	Y: 17	W: 17.3125	H: 16.71875
Offset Values:	X: 3	Y: 47.25	Ad: 23.4375

Ascii: 35	Char: [ # ]		
X: 69	Y: 238	W: 35.1875	H: 48.875
Offset Values:	X: 0.65625	Y: 48.0625	Ad: 36.71875

Ascii: 36	Char: [ \$ ]		
X: 8	Y: 48	W: 31.28125	H: 58.40625
Offset Values:	X: 2.34375	Y: 51.625	Ad: 36.71875

**Kerning Table Info** contains the kerning information for the font. Kerning data can either import at font creation time or added manually using this panel.

**Left Char** - The left character part of the pair.

**Right Char** - The right character part of the pair.

**Offset** - The offset between the left and right character.

**Kerning Table Info** (Click to collapse)

Left Char	Right Char	Offset Value
T 84	e 101	Offset -10.31
l 108	e 101	Offset 1.52
x 120	t 116	Offset 2.49

l 108	e 101	Offset 0
-------	-------	----------

Add New Kerning Pair