

HQText



Offline Documentation

www.chocolatedinosaur.com

27 September 2024



Table of contents:

- [About HQ Text](#)
 - [Introduction](#)
 - [Requirements](#)
 - [Features](#)
 - [Getting Started](#)
 - [Adding an HQText GameObject](#)
 - [Using Markup](#)
 - [Custom Fonts](#)
 - [Text Preview](#)
 - [Updating Text at Runtime](#)
 - [Creating a Typewriter Effect](#)
 - [Limitations](#)
 -  [HQText Core](#)
 -  [HQText \(UGUI\)](#)
- [HQText Core Component](#)
 - [Overview](#)
 - [Properties](#)
- [HQText Core Component](#)
 - [Overview](#)
 - [Properties](#)
- [Support](#)
 - [Product Support](#)
 - [Unity Discussions](#)



About HQ Text

Introduction

HQText is a Unity plugin developed in response to the growing need for accurate and comprehensive language support, including precise placement of diacritics, ligatures, and the proper handling of mixed bi-directional text elements. Although our primary focus has been on providing correct support for Arabic text within Unity, HQText is designed to accommodate most other languages just as effectively.

To achieve this, HQText employs a user-friendly and intuitive wrapper for Pango, an open-source library for laying out and rendering text. By default, HQText utilizes FreeType for font rendering, but it can also be configured to work with the native Windows (win32) backend.

While **TextMeshPro** is another option for supporting Arabic, it often experiences issues with diacritic placement, which can vary between fonts, and transparency artifacts arising from overlapping character ligatures. For those who find these problems unacceptable, HQText serves as an ideal alternative.

! INFO

HQText is optimised for rendering quality and language accuracy over speed.

Requirements

- Supports all Unity versions from 2019.4.0 up to Unity 6.x
- Supports all render pipelines: Built-in, URP and HDRP
- Windows only (Windows 10 & 11)

Features

- Accurate complex text rendering

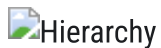
- Arabic, Persian/Farsi, Hebrew and more
- Supports ALL the render-pipelines: Built-in/URP/HDRP
- Also works in editor (no need to enter play mode)

Getting Started

Adding an HQText GameObject

It's really simple to add an HQText object to your UI Canvas:

- Right-click on your desired parent GameObject.
- Under the UI menu you will see `Text - HQText`, click to add it to your Hierarchy.



The text that is displayed, its font, and how it is laid out is set in the `HQ Text Core Component`, while the `Apply HQ Text To UI` is responsible for the rendering of the object.

Using Markup

By checking the `Use Markup` in the `HQTextCoreComponent`, Pango's support for markup will be enabled. For more information look at this [webpage](#).

Custom Fonts

To add and use a new font on your components, copy the `.ttf` or `.otf` files to the `Assets/StreamingAssets/HQFont` directory (create if necessary). Restart Unity and you should be able to select the fonts on your HQText components after startup.

Text Preview

If you have the HQText gameobject selected you will be able to see a Text Preview window at the bottom of the Hierarchy.

This will give you a detailed breakdown of your text and how HQText is building the text output. When you change settings in the components you will see them updated the text here and in the scene view.



Updating Text at Runtime

In your own code, get a reference to the `HQTextCoreComponent` that you would like to update.

Make a call to `Core.SetText`

```
public class Example
{
    public HQTextCoreComponent Text;

    void Start()
    {
        Text.Core.SetText("Your new text");
    }
}
```

Creating a Typewriter Effect

In your own code, get a reference to the relevant `ApplyHQTextToUI` component you want to apply the effect to. Exposed for your use is the `PercentageLettersToShow` property.

The number of letters in the sentence or paragraph (all text in the HQText component is taken into account) is remapped to a range of 0 - 100. You can then drive the reveal of each letter from 0 - 100 over your desired time frame.

```
public class Example
{
    public ApplyHQTextToUI Text;

    void Start()
    {
        Text.PercentageLettersToShow = 50f;
    }
}
```

Limitations

HQText does not render well if the parent canvas is set to render in `World Space`.

 **HQText Core**


 **HQText (UGUI)**

HQText Core Component

Overview

This is the core HQText component and is used to set all of the properties for the text that will be rendered. This component does not display any text in Unity. To display text another component is used, usually [HQText \(UGUI\)](#).

Properties

 Component Screenshot

Property	Type	Range	Default	Description
Input	String			The text to display.
Font Settings				
Use Markup	Bool		False	You can use markup to format the text in this component. See more about markup here.
Font Family	String			The font use for the text in this component.
Font Face	String			The font face refers to the 'style' of the font. Bold, italic, etc.
Font Size	Float		25.0	The size in points of the text.
Text Color	Color		Black	The main color of the text.

Property	Type	Range	Default	Description
Control Characters				Control characters are special characters that change how the text is rendered but aren't visible themselves.
Left to Right	Button			Left to right mark goes at the front of the string to determine directionality.
Right to Left	Button			Right to left mark goes at the front of the string to determine directionality.
Byte Order Mark	Button			Use to break Arabic letters that should join without using a space.
Arabic Letter Mark	Button			An Arabic letter mark to fool the system into thinking that there is an Arabic letter - often placed before number sequences without Arabic that should be Bidi'd to RTL.
Left to Right Mark	Button			Left to right mark control character meant to distinguish a piece of text from its surroundings.
Right to Left Mark	Button			Right to Left mark control character meant to distinguish a piece of text from its surroundings.
End Directional Isolate	Button			End Directional Isolate section.
New Line	Button			Adds a line break.
Font Size	Button			The point size of the text in this component.
Text Color	Button			The color applied to the text in this component.
Paragraph Settings				
Horizontal Alignment	Enum		Left	Where to place the text horizontally in the

Property	Type	Range	Default	Description
				text box.
Vertical Alignment	Enum		Top	Where to place the text vertically in the text box.
Line Spacing in Pixels	Float		0.0	How big to set the gap between lines. If it is 0, then it will ignore it.
Text Box Settings				
Auto Padding	Bool		True	Whether the padding should be calculated automatically. Set to false to manually enter padding values.
Padding.Left	Float		0.0	How much space there should be between the left side of the text area bounds and text.
Padding.Right	Float		0.0	How much space there should be between the right side of the text area bounds and text.
Padding.Top	Float		0.0	How much space there should be between the top side of the text area bounds and text.
Padding.Bottom	Float		0.0	How much space there should be between the bottom side of the text area bounds and text.
Horizontal Wrapping	Enum		Wrap	<p>Controls the text rendering when the text reaches the bounds of the text area.</p> <p>Options are:</p> <ul style="list-style-type: none"> Wrap - Wrap the text onto the next line. Clip - Do not wrap the text, simply clip it to the bounds of the text box. Expand - Show all the text even if it is beyond the bounds of the text area.


Property	Type	Range	Default	Description
Vertical Wrapping	Enum		Clip	<p>Controls the text rendering when the text reaches the bounds of the text area.</p> <p>Options are:</p> <ul style="list-style-type: none"> Clip - Do not wrap the text, simply clip it to the bounds of the text box. Expand - Show all the text even if it is beyond the bounds of the text area.
Advanced				
Font Backend	Enum		FreeType	<p>The font rendering system to use internally.</p> <p>Options are:</p> <ul style="list-style-type: none"> FreeType - FreeType is an open-source library for rendering fonts. Win32 - Uses the native Windows font renderer.
Auto Direction	Bool		True	<p>If true, the direction of the text will be calculated automatically based on the content. English, for instance, will be left-to-right, while Hebrew and Arabic characters will be right-to-left.</p>
Justify	Bool		False	<p>If true, each line will stretch to fill the whitespace.</p>
Resolution Multiplier	Float		1.0	<p>Multiply the resolution by this factor to supersample the text.</p>

HQText Core Component

Overview

This component displays the text that is specified in the [HQText Core component](#) in the Unity UI (UGUI / Canvas).

Properties

 Component Screenshot

Property	Type	Range	Default	Description
Core Component	HQTextCore			A reference to the core component - must be the <code>HQTextCore</code> component on this GameObject.
Reveal Letters	Float	[0..1]	1.0	A value used to drive a typewriter effect. 1.0 - all text showing, 0.0 - no text showing.
Color	Color		White	This color is multiplied by the colors generated in HQTextCore component.
Material	Material			This color is multiplied by the colors generated in HQTextCore component.
Raycast Target	Bool		True	Should this be considered a target for raycasting?




Property	Type	Range	Default	Description
Raycast Padding	Rect		0.0	Padding in pixels for the raycast.
Maskable	Bool		True	Apply masking based on mask components in the parents.
Debug				
Reference Texture	Texture			This can be used to inspect the texture that is generated by HQText. This should not be assigned to.

Support









We believe that `software is only as good as level of support and maintenance provided`.

If you're having a problem with one of our products or have a question, please get in touch via one of the methods below:

Product Support

-  [Discord Community](#)
-  [GitHub Issues/Bugs](#)
-  [Email support@chocdino.com](mailto:support@chocdino.com)

Unity Discussions

-  [UIFX Bundle](#)
 -  [UIFX - Glow Filter](#)
 -  [UIFX - Outline Filter](#)
 -  [UIFX - Extrude Filter](#)
 -  [UIFX - Drop Shadow Filter](#)
 -  [UIFX - Blur Filter](#)
 -  [UIFX - Motion Blur](#)
 -  [UIFX - Trail Effect](#)