

TEXDraw

Changelog

TEX-typing plugin for Unity

Version 5.0.0
2019 August

1 CURRENT VERSION

TEXDraw 4.4.0
June 18, 2018

2 LINKS

Asset Store Wello Soft

Look at TEXDraw menu and documentations for integrations with other assets.

3 RELEASE NOTES

4.4 - June 18, 2018:

New feature includes TEXInput allows user to edit TEXDraw in runtime. Also added more Scripting API and bugfixes to ease TEXDraw integration.

- NEW: Use double braces in TEXInput to lock/prevent modifications for specific syntax
- NEW: Support for Unity UI RectMask2D
- FIX: Improved TEXInput regex bracket matching

- FIX: Incorrect color cache invalidation
- FIX: Improved Editor material validation wording
- FIX: Threading issue when enabling TMP Integration

4.3 - March 31, 2018: The New Input System ; SUMMARY: New feature `TEXInput` allows user to edit `TEXDraw` in runtime. ; Also added more Scripting API and bugfixes to ease `TEXDraw` integration.

- NEW: `TEXInput` to allow users edit and input expressions in runtime
- NEW: `TEXInputCursor` + `TEXInputLogger` (additional components accompanying `TEXInput`)
- NEW: Sample scene for `TEXInput` demonstration
- NEW: `\vlink` to create links without input box margin
- NEW: `FillHelper.ReadMesh` + `DrawingParams.GetAreaXXX()`
- NEW: "Copy Raw Syntax" dropdown button in `TEXDraw` and `TEXDraw 3D`
- NEW: `\meta[font=xxx]` where `xxx` is font name (was only allowed as font index)
- CHANGE: `\meta[]` short tokens are deprecated
- CHANGE: `\hold` without specifying minimum size will no longer re-center the expression inside
- CHANGE: `TEXDRAW_PROFILE` renamed to `TEXDRAW_DEBUG` and more debug-only features covered behind the flag
- FIX: `\<fontid>` without style throws `IndexOutOfRangeException`
- FIX: empty `{}` creates an extra whitespace (`TEXInput`)
- FIX: Character kerning get lost if wrapped in specific attributes
- FIX: `\meta[font=xxx]` is not working properly

- FIX: TEXLink didn't listening to UI RaycastTarget
- FIX: EditorTEXFontSwapper throw null errors
- FIX: Visual Studio can't debug nor compile because of TEXDraw code
- FIX: Compiler error on Build and `TEXDRAW_TMP` (v4.2 regression)

4.2 - March 7, 2018: Extension got Extended ; SUMMARY: Fixes bugs appearing in supplements and former examples. ; Now it's possible to type Rich Tag (HTML) and Markdown in TEXDraw.

- NEW: TexSupTranslator to transpile Basic HTML + Markdown into LaTeX.
- NEW: Example scene for demonstrating HTML and Markdown syntax in TEXDraw
- NEW: Set a custom cursor when mouse hovering a link in TEXLink
- NEW: Showcase example now navigatable in gameplay using arrow keys
- CHANGE: Braces `{}` without commands will **group** the text (hence prevent wrapping)
- CHANGE: Math number is now set back as `cmr` (was using `cmmi`)
- CHANGE: `\ulink` now removes its additional margin for underline
- FIX: Incosistent wrapping using `\size` and `\style` in particular cases
- FIX: Missing normal + tangents when using Surface Shaders
- FIX: Missing 'DontSave' flags for generated mesh in TEXDraw 3D
- FIX: Remove Skies on example scenes
- FIX: Tooltips on TEXLink inspector
- FIX: Errors in TEXSupTransformFill
- FIX: Outdated NGUI Shaders and Scroll View

4.1 - February 3, 2018: Bugfixes ; SUMMARY: The code that handles layout is more readable and ; it's possible to make more consistent line height across components.

- NEW: Set minimum height and depth in each line via `TEXConfiguration`
- CHANGE: Deprecate individual 'space size' in favor to 'line space' metric in `TEXConfiguration`
- FIX: Text don't wrap (again) if wrapped inside `\font`, `\math`, `\size`, `\style`, `\color`
- FIX: `NullReferenceException` in `\hold` and scripts with empty parameters
- FIX: Sprites with unknown char will have warning instead of throwing error to console.
- FIX: Editor character map don't clamp on navigation via keyboard

4.0 - January 1, 2018: Negative Code ; SUMMARY: This version aims to increase the readability of `TEXDraw` codebase by removing old ; code and logics. This results in improvements in many areas including build size and performance.

- NEW: `\style[]` to change font bold/italic styling
- NEW: `\border[]` to add box border (can be customized)
- NEW: 'Automatic' TMP and NGUI integration, with single click
- CHANGE: Sprites now must be imported with unity's multisprite editor
- CHANGE: Minimum unity version leveled up from 5.2.1 to 5.6.0p3
- CHANGE: UV2 moved to UV3 and Tangents moved to UV2
- CHANGE: Font metadata now implented as different inheritance based on asset type

- CHANGE: `TEXPreference.main` is a field (loaded eagerly by `TEXDraw` components)
- CHANGE: XML data is removed
- CHANGE: Removed Emit logic/speedhack (unreliable)
- CHANGE: `TexSupFixTMP` and `TexSupTransformFollow` is obsoleted (now handled automatically)
- FIX: Removed unnecessary heavy payload in `TEXDraw` metadata on build
- FIX: Corrupt graphics because transformation of tangents
- FIX: Code cleanup (over +2500 lines of code get stripped)
- FIX: Faster font metadata rebuild
- FIX: NGUI shaders not included in build
- FIX: Shader warning at import for 5.6 and up

3.8 - June 17, 2017: Easebility

- NEW: `TexSupTranslator` to translate different popular text syntaxes!
- NEW: `\size[]` for reset scale to absolute 1
- NEW: `\sqrt` for `\root` equivalent
- CHANGE: Documentation is split to two: Manual & Reference
- CHANGE: New syntax & fix issues for importing custom characters
- FIX: 5.6 Compiler error with TMP integration
- FIX: `\nfrac` now have consistent margins with `FractionGapNoLine`
- FIX: Issue of Reversed wrapping justified with best fit.
- FIX: `TexSupLinkAsLayouter` doesn't scale with `canvas.scaleFactor`
- FIX: Improvements in RTL Bidirectional Detection

3.7 - May 10, 2017: Another urgent fixes

- NEW: `\meta`'s new wrap parameter
- FIX: Rebuilding font doesn't reupdate its font index
- FIX: NGUI depth issue and update to recent version
- FIX: font eufm is not listed in font stack at import
- FIX: several improvements in `TEXSupLinkAsLayouter`
- FIX: now `TEXSupLinkAsLayouter`, `TEXSupTransformFollow`, `TEXSupRTLSupport`, `TEXSupDepthEffect` is marked as stable

3.6 - May 4, 2017: Rolling for Unity 5.6

- NEW: `TEXSupLinkAsLayouter` for bind UI transforms to given links
- NEW: `\hold[x,y]` for fixed size in both width and height
- NEW: Major performance improvement in post-effects for 5.6 using new `Mesh.Get` API
- NEW: orientation marker with `\rtl` and `\ltr`. Only used when Bidirectional RTL is enabled.
- CHANGE: Now users will be warned to add `TexSupTransformFollow` if they rotate `TEXDraw` UI without tangent fixer in editor
- CHANGE: `TEXDraw`'s custom material editor will no longer appear in 5.6 because the inspector has to do it already
- FIX: Compatibility issue for Unity 5.6 and 2017.1 beta
- FIX: In 5.6, set canvas `AdditionalShaderChannels` automatically when using `TEXDraw`
- FIX: Rotation fixer of `TexSupTransformFollow` now calculates correctly when canvas is set to world and rotated
- FIX: Bidirectional RTL is now completed and can parse expression correctly

3.5 - April 22, 2017: Stability bugfixes

- NEW: Repolished example (demo) scene
- NEW: Use triple scripts `^^^` or `___` for smaller over/underscript
- NEW: `\bg` or `\vbg` for drawing background colors below given expression
- NEW: Exposed debugging tools available in `TexConfiguration`
- FIX: missing symbol `\angle` and others in both config and XML
- FIX: Bidirectional RTL feature isn't smart enough
- FIX: Regex issue with `TexSupNewLine`
- FIX: `\vhold` now can extend on fixed height correctly

3.4 - March 30, 2017: RTL & Samples Development

- NEW: RTL support as an supplement with Arabic and Persian support from external code source.
- NEW: Reversed wrapping modes for RTL text (this is optional, so need to be toggled on first on menus).
- NEW: Few example scenes with some improvements
- NEW: Benchmark tool now measures supplement and posteffects too
- NEW: `TexSupTransformFollow` contains fix for UI rotation glitches
- NEW: `TexConfiguration` for a new and consistent way to save and adjust configurations
- NEW: `TEXDraw` editor accepts Tab characters, though requires supplement to make it work properly
- NEW: Menu toggles for more ease to integrate with other assets
- NEW: Up to 25% boxing speed improvement for Non-AOT platfroms (credit to `FastReflector` by `vexe`)

- CHANGE: Added and changed some options in TexConfiguration compared from the old one.
- CHANGE: Editor changes made in TEXPreference editor for more readable code and layout
- CHANGE: Restoring a missing change from V3.0 (Color console palette)
- CHANGE: `\text` now support styling and now follows the font which choosed in the component instead
- CHANGE: scripted `\hold` with big-operator now no longer change to under-overscript for consistency
- CHANGE: Parser now ignore `\r` char so CRLF characters won't print an unwanted blank space
- FIX: Justify with right-aligned text is misaligned
- FIX: TEXPreference can't be fully recovered when accidentally deleted/corrupt
- FIX: Aspect ratio was ignored when previewing characters in preference
- FIX: MeshEffect spamming error in console when saved inside of prefab
- FIX: Unnecessary usings and files in source (good for VSCode users)
- FIX: Editor of supplements isn't available for multiple edit
- FIX: API updater is triggered when imported to Unity +5.5

3.3 - March 16, 2017: Supplement Upgrade

- NEW: A new example scene for keeping formulas in the game
- NEW: Reset button in preference context menu (suitable if you really stuck in a problem)
- NEW: Editor and additional notes for each supplement
- NEW: Editor toggle for showing supplement info, optionally (show is default)

- NEW: TEXSupDepthEffect for giving a depth effect for 3D Based UI
- NEW: TEXSupTrimSpaces for cleaning unneeded spaces in each paragraph
- NEW: (Undocumented yet) Modifiable template for creating TEXDraw objects (can be accessed via menu)
- FIX: Dirty isn't flagged at import which result in Font meta data won't be saved
- FIX: Dirty isn't flagged at editor change which result in Font meta data won't be saved
- FIX: (Only in editor) Shadow & Other built-in UI effect is not picked up automatically
- FIX: console errors isn't readable enough
- FIX: Possible error on internal part of SerializedDictionary

3.2 – Feb 18, 2017: Demo Released

- NEW: Demo (Windows build) is released for asset demonstration
- CHANGE: Integral over/underscript now aligned automatically
- CHANGE: Script without base will be right-aligned (useful for script-before-base cases)
- FIX: Infinite stack overflow when backslash typed after scripts
- FIX: 'Iconic' asset compression setting

3.1 - Feb 2, 2017: Bug Sweeping

- NEW: Basic Editor for TEXFont arbitrary assets
- NEW: Extra editor info for font import setup
- CHANGE: Importer will throw warn in console, if '-Regular' is used in font name
- FIX: Delimiter is not growing in particular cases

- FIX: Updated NGUI Scripts & Mask Shaders
- FIX: Build compilation exception & warning
- FIX: EOL consistency issue in script
- FIX: Preference will marked as 'dirty' after reimporting fonts.

3.0 – Dec 31, 2016: Repolished API

- NEW: Built-in TextMeshPro Integration
- NEW: Built-in SDF Importer and Shader for TMP Integration
- NEW: `\meta` command to apply paragraph-wide styling setting
- NEW: Rootsign of `\root` can be changed
- NEW: `\hold` command to reserve a fixed amount of space
- NEW: `\trs` and `\mtrs` to apply 3D transformation on character
- NEW: Best Fit mode in Autofit
- NEW: Local Continous and World Continous in Autofill Options
- NEW: 7 Additional supplements in package
- NEW: TEXDraw 3D accepts Rect Transform
- NEW: TEXDraw Menu items in Tools/TEXDraw
- NEW: Customizable character sets
- NEW: Benchmark tool, Font Swapper tool, and many else
- NEW: Material Assistant for quick management of different Materials
- NEW: Rebuild material now automatically choose best samples if available
- CHANGE: Improved TexDrawPerCharacterBase regex filter
- CHANGE: UV2 and tangents data is swapped for consistency

- CHANGE: Redefinition of `\clr` console color choices for consistency
- CHANGE: Deprecation of XML Data, through it is still supported to read from
- CHANGE: FontData now separated from Preference for data consistency
- CHANGE: TEXDraw no longer checks for UI mesh effect every rebuild time
- CHANGE: TEXDraw Preference now saved in Resource
- CHANGE: Improved TEXDraw 3D Editor
- CHANGE: Tables now only add outer border if necessary
- CHANGE: Renamed '4 Passes' to 'Full', and 'X Passes' to 'X Samples'
- CHANGE: Fonts that have incompatible character will be ignored
- CHANGE: Font texture will resized if rescaled to keep it crisp on screen
- FIX: Unity 5.6 Compatibility
- FIX: TEXDraw 3D double render issue in some frequent cases
- FIX: Delimiters now can consistently grow bigger if necessary
- FIX: Scripts now consistent with unbraced expressions
- FIX: Accents now only wrap more than a character if it braced
- FIX: Preference now do null checks before previewing fonts
- FIX: Small performance improvement on rebuild time

2.7 – Oct 1, 2016: Improved Editor & API

- NEW: TEXDrawSupplement API to modify text similar to UI effects
- NEW: 4 Built-in Supplements component
- NEW: Editor Highlight system for text property

- NEW: `\size[.]` will decrease font size into scripts level
- CHANGE: Splitting of parser script for ease of customization
- CHANGE: Debug Information now hiding if there is no problem
- CHANGE: One hex letter `\color` code now follows modified 4-bit colorful index
- CHANGE: Vertical delimiter now be rotated if used as horizontal delimiter
- CHANGE: font size texture is limited to 1024, prevent memory crashes
- FIX: Eliminating use of System.Linq for all runtime scripts
- FIX: Optimized Parser, minimum usage for StringBuilder
- FIX: Nested script no longer need to be braced separately
- FIX: Editor improvements for NGUI extension
- FIX: Unity 5.5 compatibility
- FIX: `\color` glitch when resizing UI
- FIX: symbol definition for `\angle`
- FIX: Horizontal Extension's width shows less than actual requirement
- FIX: Preference font preview sometimes not match with inspected font

2.6 – Aug 8, 2016: Parser Stability

- NEW: Parser now more tolerant to incomplete typos
- NEW: Braces after commands is now optional
- NEW: Font styling (Bold/Italic) included in custom font tags (`\font`)
- NEW: Accents can hold by more than one character
- NEW: Bump Lit shader
- NEW: Support for Horizontal Extension

- NEW: `\math` for turning off modified custom font tag
- NEW: offset control for `\size`
- NEW: Showcase example scene
- NEW: Added (Bonus) Editor pool check for checking pooled resources
- CHANGE: 2 user fonts now changed into more useful one
- CHANGE: behavior when imports configuration in XML files (and upgrading process)
- CHANGE: `\not` now have a very extensive customization
- CHANGE: Characters now used pixel-perfect font size according to their actual size
- CHANGE: Shader now wrapped in one `.cginc` file, improves shader readability
- FIX: Faster and less GC overhead when parsing string
- FIX: NGUI Extension Glitch 'IM HIT'
- FIX: `\text` and `\font` doesn't parsing backslashes
- FIX: radical top line floats in incorrect position
- FIX: issue with Unicode characters
- FIX: `\font` and `\color` isn't able to wrap
- FIX: nested delimiter doesn't expand it's height
- FIX: Unicode characters still do incorrect glyph sizes
- FIX: Unicode not working in inside `\font` block
- FIX: Progress bar doesn't update when Preference imports

2.5 – Jul 28, 2016: Interactive Link

- NEW: Official support for NGUI, included as an external `.unitypackage` file

- NEW: UV3 filling, unlocking many shader features and variants
- NEW: Gradient and Texture overlay shader for TEXDraw
- NEW: TEXLink component, put links over TEXDraw!
- NEW: Underline and overlined text style
- NEW: Color command variants (`\clr` and `\mclr`)
- NEW: `\size` command for having variant in sizes.
- CHANGE: Improved autofit mode
- CHANGE: straight lines now rendered last
- CHANGE: shader paths now detached from GUI:
- CHANGE: Latin symbol now follow their override font, not the default math rule.
- FIX (critical): incorrect glyph sizes on large characters
- FIX: Improved performance, only few bytes GC inprints on repaint.
- FIX: Pixel-perfect behaviour for UI
- FIX: Pathfinding of Preference is now automated
- FIX: Compatibility for Webplayer

2.4 – Jun 20, 2016: Sprite Import

- NEW: Import sprites as grid-based characters
- NEW: HTML names for `\color`
- NEW: more scene example and improved stress test scene.
- CHANGE: Shaders now split into 4 passes
- CHANGE: Increasing font limit from 15 to 31 fonts.
- CHANGE: Tabbed Symbols tab on preference

- CHANGE: Removal of Shadow lit
- FIX: Improved Performance, more less GC overhead at render.

2.3 – Apr 20, 2016: Performance Upgrade

- NEW: Autowrap and Justify alignment
- NEW: Stress Test example scene
- NEW: Font Index selection in Components
- CHANGE: API now using Resource Pooling
- FIX: Performance problem with Garbage Collector

2.2 – Apr 13, 2016: Unicodes

- NEW: Unicode Support
- NEW: `\color` command for custom text colors.
- FIX (critical): Imported fonts showing only white boxes.
- FIX: Mobile Shader now using two passes instead of one

2.1 – Apr 4, 2016: Compabilities

- NEW: TEXDraw Lit Shader (with Shadows)
- FIX: Shader Compilation for PS 3.0/4.0 and Mobile
- FIX: Problem in UI Layout and their Functionality
- FIX: Unity 5.3 Compability

2.0 – Mar 29, 2016: Package Rewrite

- NEW: 15 fonts data included in Package
- NEW: Symbols catalog is increased with total of +600 symbols
- NEW: TEXDraw Preference Editor
- NEW: Dynamic Global Preference

- 1.0 – Jan 9, 2016: First Release

Thank you for choosing TEXDraw!
Follow me @willnode on Twitter for updates.