WebGL Retina Tools 3.0

Hogbox Studios

Add Retina/HDPI resolution support to your WebGL builds with the click of a button. Eliminates the blurry textures and text seen on Macs running Unity WebGL builds. Version 3.0 adds support for Unity 2017.3 (also tested and working on 5.6, 2018.1b).

Support has also been added for dynamically adjusting the resolution of WebGL builds during runtime on any browser (not just retina screens). As well as having different default resolutions on desktop and mobile. Useful for improving performance on older devices.

A free version of this tool is available here however it only provides support of fixing development builds and includes none of the settings or extras.

Demo (built with 2017.3)

Quick Start

Create a WebGL build of your project, either release or development, then go to menu item:

> *Hbx > WebGL > Fix Last Build*

This will apply the fix to the last WebGL build you created with Unity. Alternatively you can apply the fix to an existing build by going to menu item:

> *Hbx > WebGL > Fix Existing Build*

Then selecting an existing WebGL build folder (the folder containing index.html and Build folder).

Goodbye blur :)

What's included

Hbx/WebGL/Editor/WebGLRetinaTools.cs

This is all you actually need in order to run the fix. It just needs to be in a folder named Editor as it's a Unity Editor script.

Hbx/WebGL/Plugins/hbx_WebGL_DevicePixelRatio.jslib

Native javascript WebGL plugin for controlling device pixel ratio at runtime. Not required for fix but required if you want to use the DevicePixelRatio.cs class.

Hbx/WebGL/DevicePixelRatio.cs

The script interface for the hbx_WebGL_DevicePixelRatio.jslib native plugin. Not required for the fix. Required if you want to use DynamicResolutionManager or control device pixel ratio at runtime.

Hbx/WebGL/DynamicResolutionManager.cs

An example component that automatically adjusts the device pixel ratio to control resolution based on performance. Not required for the fix.

Hbx/WebGL/Docs/Readme.pdf

This readme file you're looking at right now :p

Hbx/WebGL/Demo

Folder containing demo scene, script, texture, shader and material. This can be deleted and is just provided as an example of controlling the device pixel ratio at run time.

Unity 5.6+

As of Unity 5.6 you can use the Unity provided WebGL Templates straight out of the box. Yippie! However make sure if you're using a custom template from 5.5 or lower that you update to reflect the new template style.

Unity 5.6+ has also added beta support for WebAssembly builds, this has been tested and working but looking at the code generated it looks like things may change in the near future so watch out using this tool with WebAssembly and any version of Unity greater than 2017.3.

Settings

Version 3.0 adds a new settings window which gives you greater control over how the fix is applied. Access them by going to menu item

Hbx > WebGL > Settings
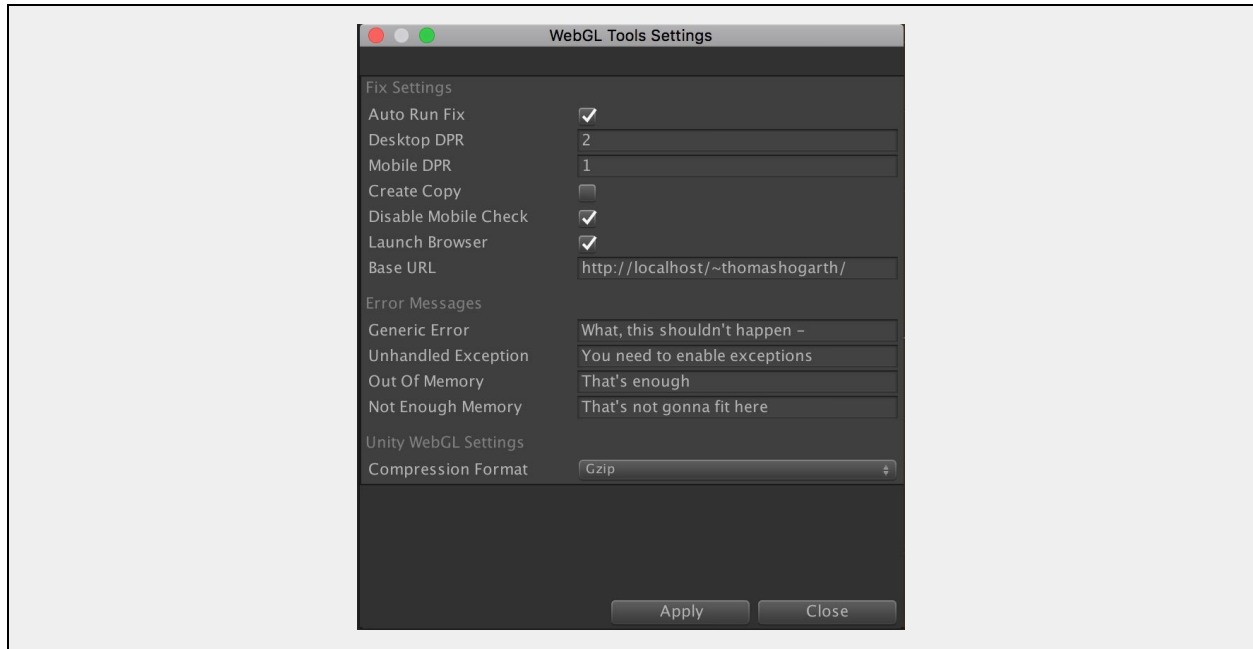
- Auto Run Fix

Selecting this option will automatically run the fix each time you create a new WebGL build.

- Desktop Scale

Scale the native device pixel ratio by this value on desktop. So a value of 1.0 is full resolution, a value of 0.5 is half resolution.

- Mobile Scale

Allows you to have a different pixel scale on mobile platforms.

- Create Copy

Should the build folder be copied and the fix applied to that copy. Handy if you want to leave your original build as is and or create multiple builds with different device pixel ratio values etc.

- Copy Append

If creating a copy this is the text we add to the end of the build folder name.

- Disable Mobile Check

By default Unity displays a popup dialog if the user runs the build on a mobile device. Selection this option will disable that popup and allow the game to run on mobile without warning. Please note though that Unity does not officially support WebGL on mobile, however simple tests have been shown to work and this fix also supports touch input on HPDI screens.

- Launch Browser

Select this option to automatically open your browser after applying a fix, for this to work you need to have a local http server running and for the build being fixed to be placed somewhere accessible by that http server.

- Base URL

If Launch Browser option is selected this is the base URL for your build. e.g. if your build was called mygame and you can goto http://localhost/mysites/mygame to view it, then just enter http://localhost/mysites/.

- Error Messages

Unity has some default error messages for things like running out of memory, prompting the user that mobile isn't supported etc etc. This section allows you to override some of those warning messages. Leaving any field blank will use Unity's default message.

- Compression Format

Unity's own compression setting for WebGL builds. Release WebGL builds support compression using GZip or Brotli, at present only GZip is supported in WebGL Retina Tools. Some work has been done to add support for Brotli but this is still not complete in version 3.0. This setting is just here for convenience.

Extras

WebGL Retina Tools now contains a few additional classes to allow you to control the device pixel ratio at run time.

- DevicePixelRatio.cs

This class works along side the hbx_WebGL_DeivcePixelRatio.jslib native plugin. It allows you to get/set the device pixel ratio used by the browser during runtime. For convenience you can also set the device pixel ratio as a scale of the native dpr.

- DynamicResolutionManager.cs

A simple example component that dynamically adjusts the device pixel ratio based on performance (the average frames per second). Attach this behaviour to a gameobject in your scene and it'll do the rest automatically. Only really provided as an example for users to work off of.

Important Notes:

- You **Must** have a css style on the canvas container controlling it's width and height. If you don't have one the canvas will keep growing.
- If you need your view to resize as the browser window resizes enable "Run in Background" in the player settings. Otherwise the view dimensions will not update until you refocus the view.
- This script depends on find and replace so is only guaranteed to work with the versions of Unity it has been tested against (5.6, 2017.3, 2018.1b). Hopefully Unity will eventually support this out of the box.
- This script will **Not** add Retina support to standalone desktop builds, only WebGL.
- You need to reapply the fix each time you create a new build, unless Auto Run Fix is active.