

## Browser TUIO configuration

This document has been superseded and remains for reference only. Since April 2019 see [new TUIO browser interface document](#)

Outdated!

We have been asked on a few occasions how you might utilise multi-touch gestures in a browser under Mac OS X. We are aware that when using UPDD Gesture software the gestures cannot be used to control touch-enabled web pages but are performed on the browser dialog.

The problem is that OS X doesn't pass its system-level touch events to web browsers as HTML5 / javascript touches. So while these pages will work with an iPad (and similar) it can't work in OS X, even when using Apple's magic trackpad.

However, there is a solution albeit it is a bit of a convoluted hack.

Please also note that the steps listed below for Chrome no longer work with latest chrome versions. It appears that Chrome dropped support for NPAPI plugins with version 42. NPAPI plugins are the old variety of web browser plugins that have been used for things like Flash for some time (the acronym stands for 'Netscape Plugin Application Programming Interface' to give an idea for how long these have been around). However one of the steps for getting TUIO touches in a desktop browser involves installing an NPAPI plugin called npTuioClient.

The weak link here is pretty clearly npTuioClient. It's a plugin that's been around for a while and appears to no longer be supported or updated and exists because, before our current modern versions of html and javascript came along, there wasn't any other way to get TUIO touches in a browser. That's no longer the case.

If you are developing your own webpage and would like to support touch via TUIO in addition to native javascript touches try using a library like this:  
<https://github.com/fe9lix/Tuio.js>

We haven't personally tested this but at first glance it looks like it should work. In theory using that in a webpage in conjunction with UPDD TUIO would allow the page to receive all touches from the UPDD.

Having done further investigations we think we could develop a javascript library that receives TUIO touches and translates them into native javascript touches. The library could

## Browser TUIO configuration

be packaged as an extension for Safari, Firefox, and Chrome that enables any webpage to receive TUIO touches as though they were native javascript touches.

Extending some work on browser multi-touch (and the resultant magictouch.js developed by Boris Smus) we have been able to implement a 'hack' such that all web pages are slightly modified so that touches will work. This 'hack' converts TUIO touches to HTML5 touches and we have solutions for Chrome and Firefox as documented below.

Before setting this up in either Firefox or Chrome, it's necessary to first install the npTuioClient plug-in. Download link and installation instructions:

<https://github.com/fajran/npTuioClient>. In some of our tests we had issues with this plugin and used the latest one at <https://github.com/fajran/npTuioClient/issues/8> and we recommend this one is used on most recent Mac OS X versions (we tested on 10.6.1 and 10.8.4)

Essentially all that's necessary is to download npTuioClient.bundle (the older one creates a file called npTuioClient.plugin, the new one a file npTuioClient.bundle) and copy it into "~/Library/Internet Plug-ins". This plug-in will be available for all browsers. We recommend quitting any open browsers before installing it.

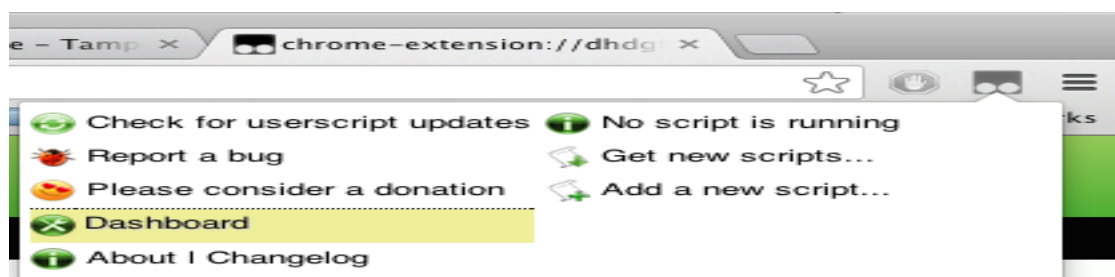
### Instructions for Chrome

1. Launch Chrome, go to this URL to install the extension Tampermonkey:

<https://chrome.google.com/webstore/detail/tampermonkey/dhdgffkkebhmkfjojejmpbldmpobfkfo>

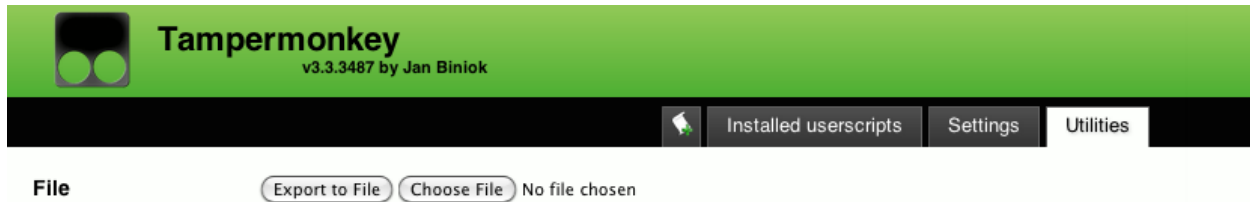
2. Add Tampermonkey to Chrome

3. A button will appear in the upper right portion of the window that looks like a box with two dots on the bottom of it. Click this and pick "Dashboard":



## Browser TUIO configuration

4. Click the "Utilities" tab in the Tampermonkey settings.
5. Click "Choose File" (download first from link below)



6. Select "[tuioTampermonkeyScript.txt](#)", Click OK in the dialog that appears to continue.

### Instructions for Firefox

1. Launch Firefox, go to this URL to install the extension Greasemonkey:

<https://addons.mozilla.org/en-US/firefox/addon/greasemonkey/>

2. Add it to Firefox
3. Restart Firefox
4. Drag the file "[addtuio.user.js](#)" (click to download) on to the Firefox window. It will prompt you to install a Greasemonkey script, click 'Install'.

After running these instructions either browser should start working with TUIO touches. If UPDD TUIO is running, UPDD touches should be received by the web browser. Here's a few web pages we've been using for testing:

<http://stickmanventures.com/labs/demo/onscreen-keyboard-html5-canvas-multitouch/>

<http://tomicloud.com/2012/03/multi-touch-demo>

<http://dev.globis.ethz.ch/jqmultitouch/1.0/behaviours.html>

This 'hack' does have some limitations:

## Browser TUIO configuration

- Webpages that try to detect if a browser supports multi-touch may not work, depending on how they go about doing that. (There are several methods, some will fail, others will work.)
- For the touches to properly line up with a web page, it's necessary for the browser to be running in full screen mode, without toolbars. Chrome's "presentation mode" accomplishes this. It may be possible to remove this limitation if required.
- The touch events this method produces may not be exactly the same as touches on an iPad or Android device, so some webpages may not work as expected.

If you have a specific webpage that does not work with this hack we may be able to improve the scripts to suit your requirements. Please contact us for further assistance

***This hack is used at your own risk as we cannot guarantee that there may be some browser issues that arise from implementing these extensions and TUIO scripts. Any feed back much appreciated.***

Touch-Base Support

<http://support.touch-base.com/Documentation/50222/Browser-TUIO-configuration...>